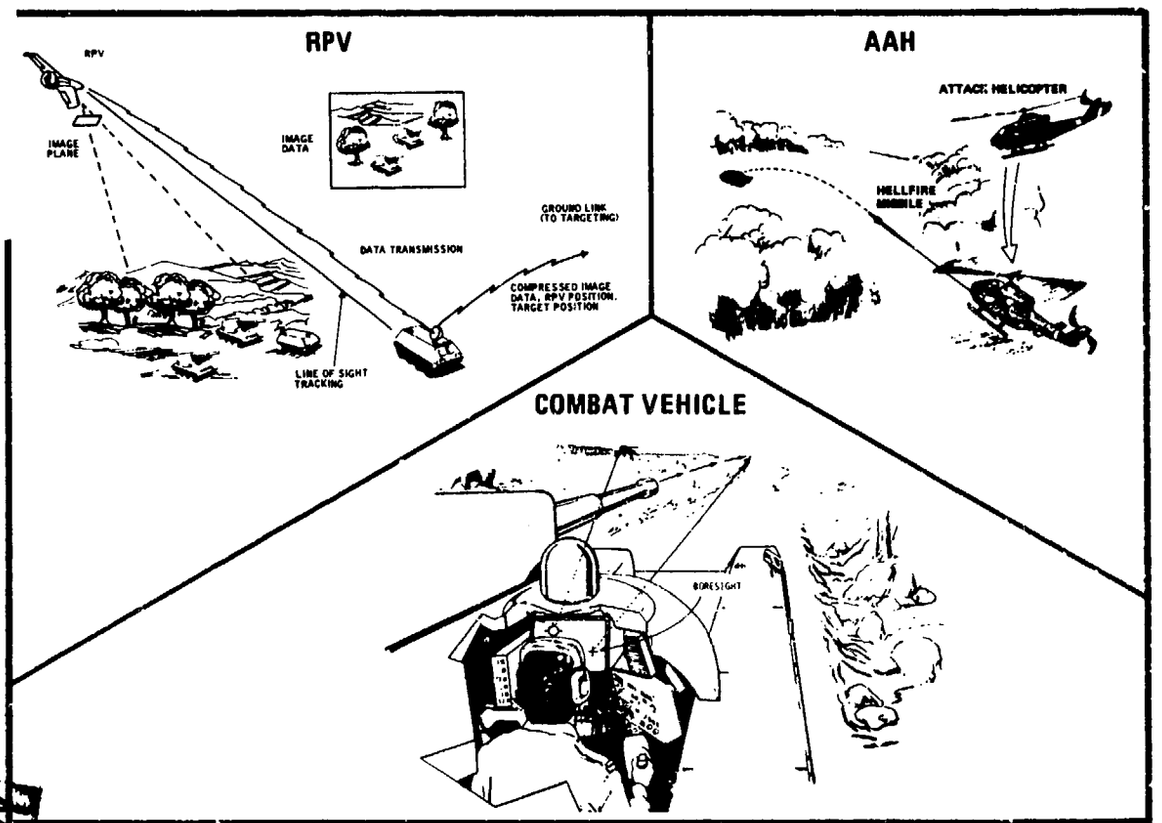
LEVEL

AD A083640

ADVANCED TARGET TRACKING CONCEPTS



FIRST QUARTERLY PROGRESS REPORT

28 SEPTEMBER to 28 DECEMBER 1979

Prepared for
UNITED STATES ARMY
Night Vision and Electro-Optics Laboratory
Fort Belvoir, Virginia 22060

DDC FILE COPY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DTIC ELECTE APR 28 1980 S D

A

Honeywell

SYSTEMS & RESEARCH CENTER
2600 RIDGWAY PARKWAY MINNEAPOLIS, MINNESOTA 55413

80 4 25 034

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (WHEN DATA ENTERED)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NUMBER AD-A083 640	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (AND SUBTITLE) (6) ADVANCED TARGET TRACKER CONCEPTS.		5. TYPE OF REPORT/PERIOD COVERED (9) Quarterly Progress Report, no. 1, 28 Sep [redacted] — 28 Dec 79.
		6. PERFORMING ORG. REPORT NUMBER (14) 80SRC6
7. AUTHOR(S) (10) P.M. Narendra [redacted] B.L. Westover		8. CONTRACT OR GRANT NUMBER(S) (15) DAAK70-79-C-0150
9. PERFORMING ORGANIZATIONS NAME/ADDRESS Honeywell [redacted], Systems & Research Center, 2600 Ridgway Parkway Minneapolis., Minnesota 55413 (11)		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME/ADDRESS Night Vision and Electro-Optics Laboratory Fort Belvoir., Virginia 22060		12. REPORT DATE Jan [redacted] 80 (12) 77
		13. NUMBER OF PAGES 75
14. MONITORING AGENCY NAME/ADDRESS (IF DIFFERENT FROM CONT. OFF.)		15. SECURITY CLASSIFICATION (OF THIS REPORT) UNCLASSIFIED
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (OF THIS REPORT)

Distribution of this document is unlimited.

17. DISTRIBUTION STATEMENT (OF THE ABSTRACT ENTERED IN BLOCK 20, IF DIFFERENT FROM REPORT)

18. SUPPLEMENTARY NOTES

The project monitor at NV&EOL is CPT Benjamin Reischer.

19. KEY WORDS (CONTINUE ON REVERSE SIDE IF NECESSARY AND IDENTIFY BY BLOCK NUMBER)

Pattern recognition, Multiple targets, Target tracking, Target cueing, Target screening, Image processing, Scene analysis, Artificial intelligence

20. ABSTRACT (CONTINUE ON REVERSE SIDE IF NECESSARY AND IDENTIFY BY BLOCK NUMBER)

Conventional target tracking approaches rely on numerical correlation over successive frames on a window around the target. They are therefore sensitive to partial obscuration and changes in target and background appearances. Furthermore, multiple-target tracking requires replication of the hardware.

In this report, we present the development of a multiple-target tracking approach based on a dynamic scene model derived from the analysis of a time

→ next Page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 55 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (WHEN DATA [redacted])

402 349

cont

sequence of imagery. Simulation results demonstrate multiple-target tracking in cluttered backgrounds and in imagery from fast-moving platforms. The approach can be implemented as an integral part of the Honeywell target screener system.

→ to p. 1

TABLE OF CONTENTS

		Page
SECTION I	INTRODUCTION	1
	Summary of Progress	3
	Report Organization	4
SECTION II	SYSTEM OVERVIEW	5
	Motion-Enhanced Segmentation Schemes	9
	Object-Matching Techniques	9
	Scene Model	10
	Target/Background Signature Prediction Techniques	11
	Advanced Algorithms for Target Detection/ Recognition/Prioritization and Critical Aimpoint Selection	11
SECTION III	OBJECT-MATCHING SCHEMES	12
	A Simple Object-Matching Scheme	15
	Fast Silhouette-Matching Algorithm	22
SECTION IV	SCENE MODEL	37
	Three-Parameter Estimation of Scene Motion	39
	Five-Parameter Estimation of Scene Motion	43
	Six-Parameter Estimation of Scene Motion	44
SECTION V	SYSTEM SIMULATION	46
SECTION VI	DATA BASE	58
	NV&EOL "Fort Polk" Selections	59
	High-Performance Aircraft Selections	59
	LOHTADS Data Base	59
	PATS Training Data Base	65
	NV&EOL TV Tapes	65
SECTION VVI	PLANS FOR THE NEXT REPORTING PERIOD	67
	PATS Simulation Software Transfer	67
	Object Matching Algorithms	67
	Scene Model	68
	Background Prediction	
	Advanced Target Recognition Techniques	69
	Homing Techniques	69
	Data Base	69
REFERENCES		70

LIST OF FIGURES

Figure		Page
1	Typical Army Scenarios Which Require Advanced Multiple-Target Tracking Through High Clutter	2
2	Overview of the Advanced Target-Tracking Approach	7
3	Advanced Target Tracker Program Overview with the Key Function	8
4	Pair of Segmented Successive Image Frames Illustrating the Key Issues Encountered in Object-Matching--Occlusion and Inconsistent Segmentation	14
5	A Simple Object-Matching Algorithm	17
6	A Sequence of FLIR Frames	19
7	Results of Segmentation and Simple Object-Matching	20
8	Centroid Test for FSMA	24
9	FSMA Flow Chart	25
10	Silhouette-Matching Example	27
11	Real-World Silhouette-Matching: Example 1	30
12	Real-World Silhouette-Matching: Example 2	32
13	Real-World Silhouette-Matching: Example 3	33
14	Two FLIR Images, From a Sequence, From a Moving Platform	34
15	Result of Segmentation and Silhouette-Matching	35
16	A Sensor Fixed in Space	39
17	Effect of Rotation About ϕ_1	40
18	Effect of Rotation About ϕ_2	40

LIST OF FIGURES--Concluded

Figure		Page
19	System Simulation Block Diagram	47
20	System Simulation Flow Chart	49
21	Three FLIR Images	51
22	Unaligned Object Outlines	52
23	Object Outlines After Alignment	52
24	Magnified View of Aligned Clutter Object Outlines	54
25	Magnified View of Moving Tank Outline	55
26	Two Successive FLIR Frames From a Stationary Target	55
27	Object Outlines Superimposed, After Object Matching and Compensating for Scene Motion	56
28	Magnified View of Stationary Tank in Figure 9, After Frame Alignment	57
29	Magnified View of Moving APC in Figure 9	57
30	Magnified View of Moving Tank in Figure 9	57
31	Example of "Fort Polk" Selections (Digitized)	60
32	High-Performance Aircraft Selections	62
33	LOHTADS Examples	64
34	PATS Training Data Base	66

SECTION I
INTRODUCTION

This is the first Quarterly Progress Report on "Advanced Target Tracker Concepts," NV&EOL Contract No. DAAK70-79-C-0150. It reports the results of the work performed between 28 September and 28 December 1979.

Tracking targets in video from TV and FLIR sensors is essential for fire control in weapon systems using electro-optical target acquisition. ~~Figure 1 shows~~ typical Army applications are shown: a remotely piloted vehicle (RPV); an advanced attack helicopter (AAH); and a combat vehicle (CV). Target tracking in these applications yields the target position for accurate pointing of a laser designator for a smart munition, such as Hellfire and Copperhead, or for fire control of conventional weapons.

Currently fielded trackers rely on numerical correlation over successive frames on a window around the target to be tracked. Several variations of the basic correlation scheme exist, and a detailed survey can be found in ref. 1. Conventional trackers are capable of tracking a manually acquired single target in relatively clutter-free backgrounds. But target-tracking requirements in the increasingly sophisticated weapon systems have grown beyond the capabilities of the current correlation trackers.[1]

---

[1] Reischer, B., "Assessment of Target Tracking Techniques," Proc., SPIE, pp. 67-71, Vol. 178. Smart Sensors, 1979.

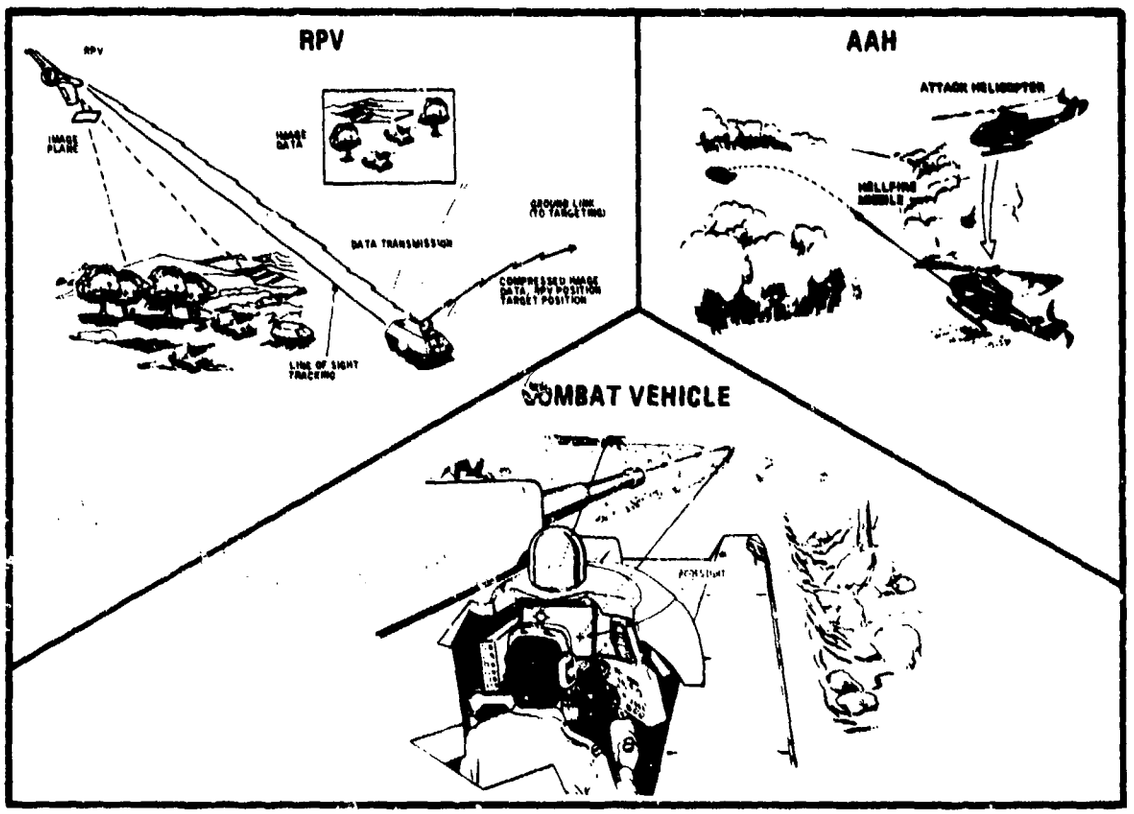


Figure 1. Typical Army Scenarios Which Require Advanced Multiple-Target Tracking Through High Clutter

Among these requirements are: 1) automatic target detection (acquisition), recognition, and prioritization; 2) simultaneous tracking of multiple targets in the presence of high clutter, obscuration, and low contrast; and 3) critical aimpoint selection.

In this program, Honeywell Systems and Research Center is developing an advanced target-tracker approach based on dynamic scene

analysis. This approach integrates the target screening functions with target tracking to provide automatic acquisition and multiple-target tracking capability with minimum additional hardware. The advanced target tracker will feature the following functional capabilities:

- Acquires targets automatically
- Tracks multiple targets (in and out of the field of view)
- Tracks partially occluded targets
- Recognizes and assigns priorities to all objects
- Performs critical aimpoint selection
- Tracks in low-contrast, high-clutter backgrounds

SUMMARY OF PROGRESS

Several significant accomplishments toward the program objectives were made in this reporting period:

- A simple feature-based, object-matching algorithm was developed, implemented, and tested on digitized FLIR imagery.

- A fast silhouette-based, object-matching algorithm was developed, implemented, and tested. This algorithm is capable of finding precise (to the pixel) positions of corresponding objects, even in the presence of segmentation noise and target obscuration.

- Dynamic models of sensor/platform motion were derived, and several alternatives were evaluated and sucessfully demonstrated.

- An integrated-system simulation incorporating both object-matching algorithms and the sensor/platform model was implemented and demonstrated on two sequences of FLIR images with multiple targets from moving and stationary platforms. The results demonstrate precise tracking capability with multiple targets in high-clutter scenes, as well as detection of minute target motion in the presence of extreme sensor motion--for moving-target detection.

- A preliminary data base of two sequences (10 frames each) of FLIR imagery was digitized to evaluate the algorithms and the current system simulation. The sequences represent high clutter, partial obscuration, and multiple moving targets from stationary and moving platforms.

- Prototype Automatic Target Screener (PATS) software was partially converted to the NV&EOL image-processing system to facilitate the installation of the software at NV&EOL in the next reporting period.

REPORT ORGANIZATION

The remaining sections of this report are organized as follows:

- System Overview (Section II)
- Object-Matching Algorithms (Section III)
- Scene Model (Section IV)
- System Simulation (Section V)
- Data Base (Section VI)
- Plans for the Next Reporting Period (SecionVII)

SECTION II
SYSTEM OVERVIEW

This section presents the system overview to introduce the program approach and terminology. The subsequent sections report the progress accomplished in this reporting period against the program objectives described in this section.

The performance goals of the advanced target tracker include:

- Automatic target detection (acquisition), recognition, and prioritization.
- Simultaneous tracking of multiple targets in the presence of clutter, obscuration, and low contrast.
- Critical aimpoint selection.

An obvious approach to add the automatic target detection (acquisition) and recognition functions to a tracker system would be to use a target screener (cuer).[2,3] The target screener would detect

---

[2]Soland, D. and Narendra, P., "Prototype Automatic Target Screener," Ibid, pp. 175-184.

[3]Soland, D., et al., "Prototype Automatic Target Screener, Goals and Implementation," U.S. Army Missile Command Workshop on Imaging Tracker and Autonomous Acquisition," November 1979.

and recognize the target and "hand off" to a separate conventional correlation tracker by supplying the target position to center the tracker window. Indeed, this distinct cuer and tracker approach has been suggested.[4] However, while the target screener (cuer) does provide the automatic target acquisition capability, this approach suffers from essentially all the drawbacks of conventional trackers with manual acquisition; that is, multiple-target tracking requires multiple copies of the correlation tracker hardware, and the tracking performance through clutter and obscuration is still limited by the correlation tracker.

The advanced target-tracker approach being developed in this program is an integrated target-screening/tracking approach which can provide automatic acquisition and multiple-target tracking through low signal-to-noise and high clutter conditions. This is done with minimal additional hardware to a target screener.

Figure 2 is an overview block diagram of the basic approach, which builds upon the scene analysis functions performed by the target screener to perform the advanced tracking function. The basic premise is very simple: the target screener segments and classifies significant objects (targets and clutter) in real time on a frame-by-frame basis. The symbolic descriptions of the objects in each frame are used to find the corresponding objects in previous frames encompassing the history of the scene. Once the corresponding object matches are made, the scene model, which includes the sensor and object dynamics as well as the target classes, is updated. Because we are keeping track of the positions of all the objects in the scene (targets and clutter), we can predict impending occlusion and future target/background signatures. Multiple-target

[4]Willet, T. and Raimondi, P.K., "Intelligent Tracking Techniques - A Progress Report," Proc., SPIE, pp 72-75, Vol. 178, Smart Sensors, 1979.

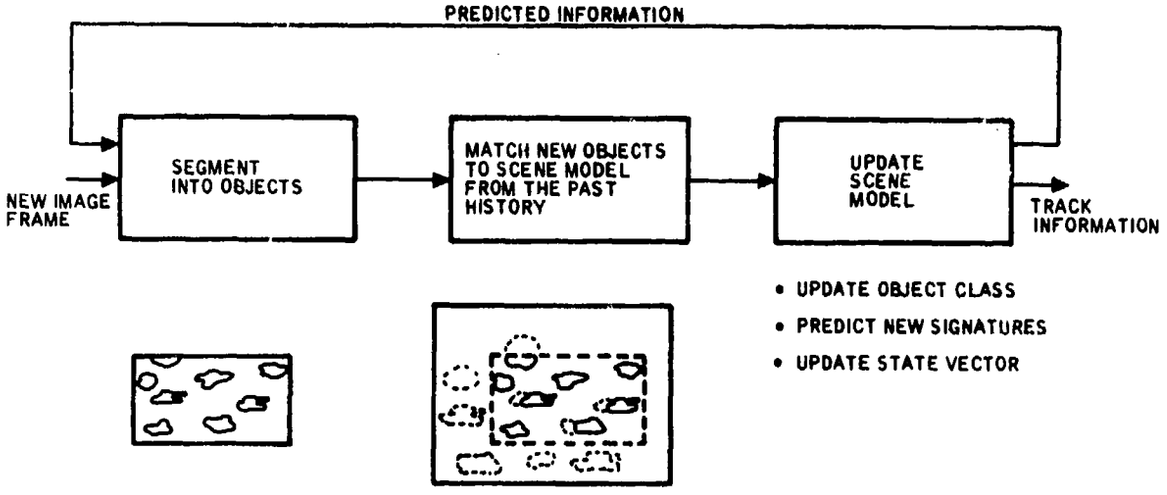


Figure 2. Overview of the Advanced Target-Tracking Approach

tracking, of course, comes free. The scene model, based on the past history of the scene, can extend beyond the current field of view. This allows reacquisition and tracking of targets which wander in and out of the field of view because of sensor platform motion.

Not only does this approach exploit the segmentation of objects from the target-screening function, but it actually improves the target detection and recognition performance over single-frame screening/cueing. First, the single-frame classification decisions of the corresponding objects are accumulated over several frames to compute an a posteriori estimate of the classification. This improves the ratio of probability of correct classification to false alarm by an order of magnitude. Second, target motion relative to the scene is detected because of the precise matching of object positions inherent in the approach. This is especially advantageous in the presence of extreme platform motion, as with

an unstabilized platform on an RPV. Motion cues can enhance the long-range target detection capability in scenarios in which a significant fraction of the targets are moving. Conventional moving-target indicator (MTI) approaches fail in these unstabilized moving-platform applications.

A complete block diagram of the major functions necessary to implement the advanced target-tracker concept is shown in Figure 3.

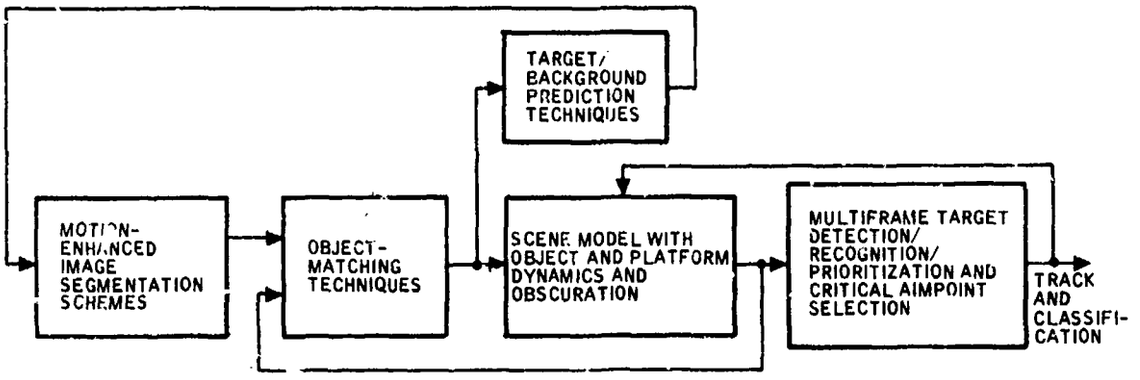


Figure 3. Advanced Target Tracker Program Overview with the Key Functions

These functions represent the major thrusts of the current program. They are:

- Efficient motion enhanced scene segmentation schemes
- Object-matching techniques capable of precise matching of objects in the new frame to the scene model derived from previous frames
- A scene model capable of characterizing object and platform dynamics, target/background signatures, and object occlusion

- Target/background signature prediction techniques to improve the probability of target acquisition in low signal-to-noise ratios
- Advanced target detection/recognition/prioritization and critical aimpoint selection algorithms, which can exploit the dynamic multiframe information

Each of these functions is introduced briefly below.

MOTION-ENHANCED SEGMENTATION SCHEMES

Object extraction (segmentation) in the integrated tracker/screener application is unique in that each frame is being analyzed in the context of the previous frames. However, conventional techniques for image segmentation do not use information from the previous frames to segment objects in the current frame. The current program uses the Honeywell Prototype Automatic Target Screener (PATS) segmentation algorithm as the baseline segmentation approach. This segmentation technique will be modified to incorporate the a priori predicted information on object/background signatures for more optimal segmentation. This effort will be directed at incorporating the interframe knowledge of the target shape and intensity signatures, as well as background characteristics expected at various locations in the frame as predicted by the scene model below.

OBJECT-MATCHING TECHNIQUES

The key to successful tracking of multiple targets in our approach depends on precise matching of segmented objects in the current

frame with the scene model derived from previous frames. This allows the precise tracking of the object positions for laser designation or for hand-off to other subsystems. Key issues in object-matching techniques are unambiguous matching in the presence of occlusion and segmentation differences due to noise, and computational efficiency of the algorithm. The development of object-matching techniques has been one of the major thrusts of this program in this reporting period. A simple object-matching technique has been developed for preliminary matching, to be followed by a sophisticated yet fast silhouette-based object-matching technique which yields the precise position of the target in successive frames.

SCENE MODEL

The scene model is a collection of information from previous frames, against which the new frame can be compared. It consists of the object shapes and positions from previous frames, the object dynamics (object positions and velocities), and the sensor/platform motion dynamics (position and velocity). In addition, the scene model must be capable of predicting occlusion and signature change of a target as it approaches occluding objects. The development of the scene model is an evolutionary process. The implementation of the scene model at this time includes the estimation of the sensor position based on the positions of corresponding stationary objects found by the object-matching algorithms. This scene model successfully aligns frames which have been transformed because of sensor/platform motion and is capable of discriminating a minute relative target motion in the presence of extreme sensor/platform motion.

TARGET/BACKGROUND SIGNATURE PREDICTION TECHNIQUES

The purpose of this effort is to use the multiframe information on the target position and dynamics to predict the target shape, intensity signatures and position, and background characteristics expected at various locations in the frame. This information is used by the motion-enhanced segmentation scheme to improve the target acquisition probability in the presence of low signal-to-noise ratios and high clutter.

ADVANCED ALGORITHMS FOR TARGET DETECTION/RECOGNITION/ PRIORITIZATION AND CRITICAL AIMPOINT SELECTION

These functions are performed in current target screeners on a frame-by-frame basis. The purpose of this task is to use the multiframe information to improve the performance of these functions in the integrated system. This improvement will be brought about in two ways. First, by accumulating multiframe decisions of corresponding objects to improve the classification accuracy over single-frame analysis. The second improvement to the classification function takes advantage of the fact that moving objects will, in general, be targets. Thus, the problem of target recognition can be improved by a moving-target detection algorithm. In this reporting period, we have demonstrated the feasibility of moving-target detection in the presence of substantial sensor/platform motion using these techniques. Critical aimpoint selection is an important function required in terminal homing munitions and its implementation with syntactic techniques will be addressed in subsequent reporting periods.

SECTION III
OBJECT-MATCHING SCHEMES

As noted in Section II, object matching is performed on the output of object segmentation. Its purpose is to find the positions of corresponding objects in successive frames. It is therefore key to track the object positions as the sensor and the targets move from one frame to the next. Object matching not only finds the positions of the moving targets in successive frames but also identifies corresponding stationary (clutter) objects in the scene. The positions of these corresponding stationary objects are input to the scene (sensor/platform) dynamics model for computing the platform motion, as discussed in the next section.

The key issues to be addressed in the development of succesful object-matching algorithms are:

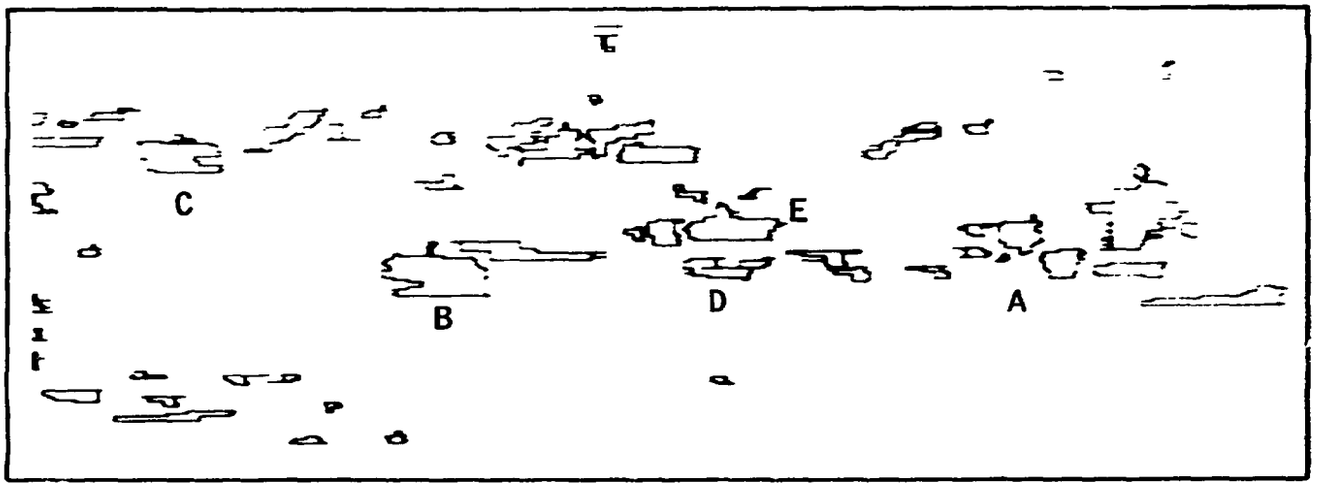
- Occlusion
- Inconsistent segmentation

The principal effect of object occlusion (partial or total) is that the object shape descriptors change, making it difficult to match objects in successive frames. For example, when a target goes behind concealing background, the leading edge of the target disappears. Inconsistent segmentation usually results from poor signal-to-noise ratio and segmentation algorithm anomalies. For example, objects extracted in one frame may not appear in the subsequent frames; an object extracted as one segment in one

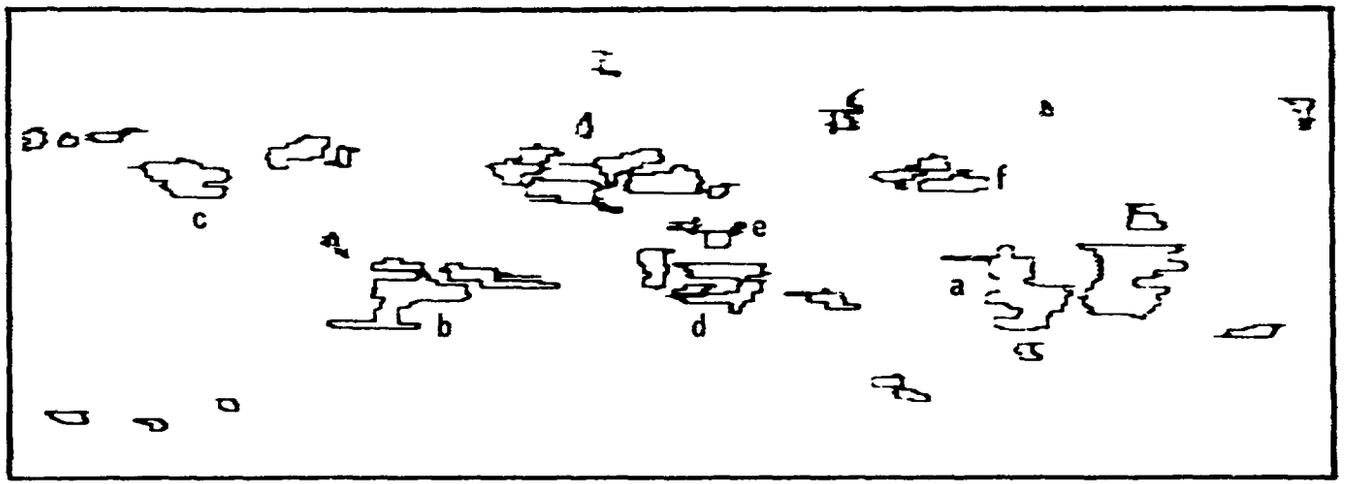
frame may appear as multiple segments in the subsequent frames or vice versa. The outlines of the segments extracted may change shape drastically because of change in target/background contrast from one frame to the next.

These issues are illustrated in the example in Figure 4, which shows two successive frames (240 msec apart) from a sequence of FLIR imagery of a scene containing multiple moving and stationary targets (tanks and APCs). The two hotspots represented by A in Figure 4a have been merged into one segment in Figure 4b, as the two tanks move close together so as to partially occlude each other. Other objects, such as object B, have drastically changed their shapes. An even more challenging example is seen with objects D and E. Object d in Figure 4b is a combination of parts of objects D and E in Figure 4a. Object e in Figure 4b is a combination of parts of objects D and E in Figure 4a. This example illustrates that one-to-one, many-to-one, one-to-many, and many-to-many object matches will have to be found. Furthermore, not all objects have corresponding matches in successive frames. For example, object f in Figure 4b does not have a counterpart in Figure 4a.

It is not sufficient to identify corresponding objects in successive frames; it is necessary to find their precise positions. This is especially important in the light of inconsistent segmentations and target obscuration which can cause the shape of the object to change drastically from one frame to the next. To illustrate this point further, consider two objects with drastically different shapes in successive frames. After performing object association, if we use the positions of the centroids of the object in each frame as the apparent position



a) First Frame



b) Next Succeeding Frame

Figure 4. Pair of Segmented Successive Image Frames Illustrating the Key Issues Encountered in Object-Matching-- Occlusion and Inconsistent Segmentation

of the object in the field of view, there would be an apparent motion (jitter) in the position of the target, even if the target has not moved, because the centroid positions change because of the change in target shape. Therefore, the object-matching technique must determine precisely how much the objects have moved from one frame to the next.

Two distinct techniques for performing the object matching have been developed. One is the simple feature-based object-matching technique which finds corresponding objects based on simply derived object descriptors such as contrast, shape, etc. It succeeds in finding initial matches of corresponding objects with consistent segmentations. To handle inconsistent segmentations and to obtain precise positions of objects in successive frames, a fast silhouette-matching algorithm has been developed. This algorithm works on the segmented outlines of the objects and rapidly converges to a precise registration of objects in successive frames. The nature of this algorithm allows it to handle inconsistent segmentations which result in one-to-one, one-to-many, many-to-one, and many-to-many object matches, as discussed below.

A SIMPLE OBJECT-MATCHING SCHEME

A simple feature-based object-matching scheme was developed to rapidly find those objects which have nearly identical segmentations in two successive frames. Since feature matching is the first step in the object-matching process, it will operate on two frames which have not been aligned to account for sensor motion; the algorithm should not be sensitive to improper frame

registration. Object matches found by this scheme are used to estimate the transformation from the previous frame to the current frame. Therefore, the matching algorithm must provide an adequate number of accurate matches for this estimation. The following paragraphs describe the algorithm.

The PATS segmentation yields object outlines and associated feature vectors. The feature-based algorithm attempts to match objects between frames by comparing a subset of PATS features. The subset contains the following features:

- Object centroid position
- Object contrast
- Object area

These three features are used to find a corresponding object in the current frame for each object in the previous frame. This matching process is illustrated in the flow chart in Figure 5 and described in the following paragraphs.

The object centroid position is used to limit the size of the search region in the current frame. Only those objects in the current frame which are within a given number of pixels, N, of the object position in the previous frame are considered for matching. This, of course, limits the amount of frame motion that the algorithm can withstand. However, the search region can be made large enough, say one-eighth the frame size, to accommodate extreme sensor motion.

A distinguishing characteristic of an object is whether it is hotter or colder than the local background. Therefore, the object

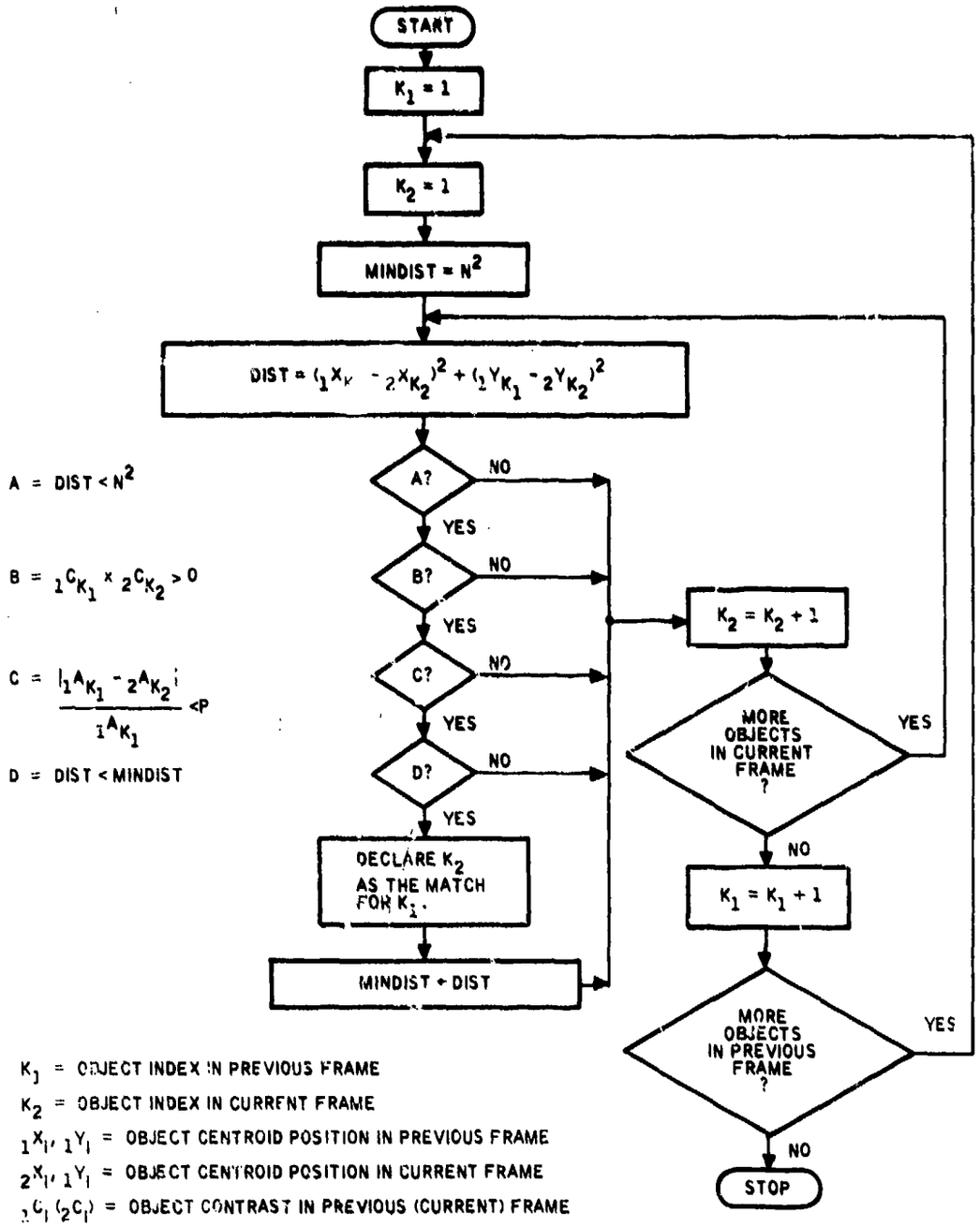


Figure 5. A Simple Object-Matching Algorithm

contrast was chosen for use in the simple matching algorithm. The contrast test limited the search to those objects which had the same sign on the contrast. This ensured matching hot objects with hot objects and cold objects with cold ones.

The centroid and contrast tests verify that the object locations and intensities are similar. The comparison of object areas tests the relative sizes of the objects to be matched. Only those objects which differ by less than some percentage, P, of the area of the object in the previous frame are considered in the matching process.

For a given object in the previous frame, several objects from the current frame can pass all three tests. If this occurs, then the object which is closest to the object position is chosen as the match. This method will provide accurate matches when the frame displacement is small or when the two frames are approximately aligned. The approximate alignment can be derived from a history of the platform motion or, as currently implemented in the system simulation, by using the simple matching algorithm to find matching objects and compute an approximate transformation. This sequence is iterated until no new matches are found.

The results of applying the simple object-matching algorithm to a tactical scene are shown in Figures 6 and 7. Figure 6 is a sequence of four FLIR frames approximately 0.2 second apart. A column of moving tanks and APCs is seen in the background, while stationary tanks are seen in the foreground. Figure 7 shows the results of segmentation and object matching on this sequence. Objects bearing the same label have been matched between scenes. Objects not matched have new labels. Numerous object matches

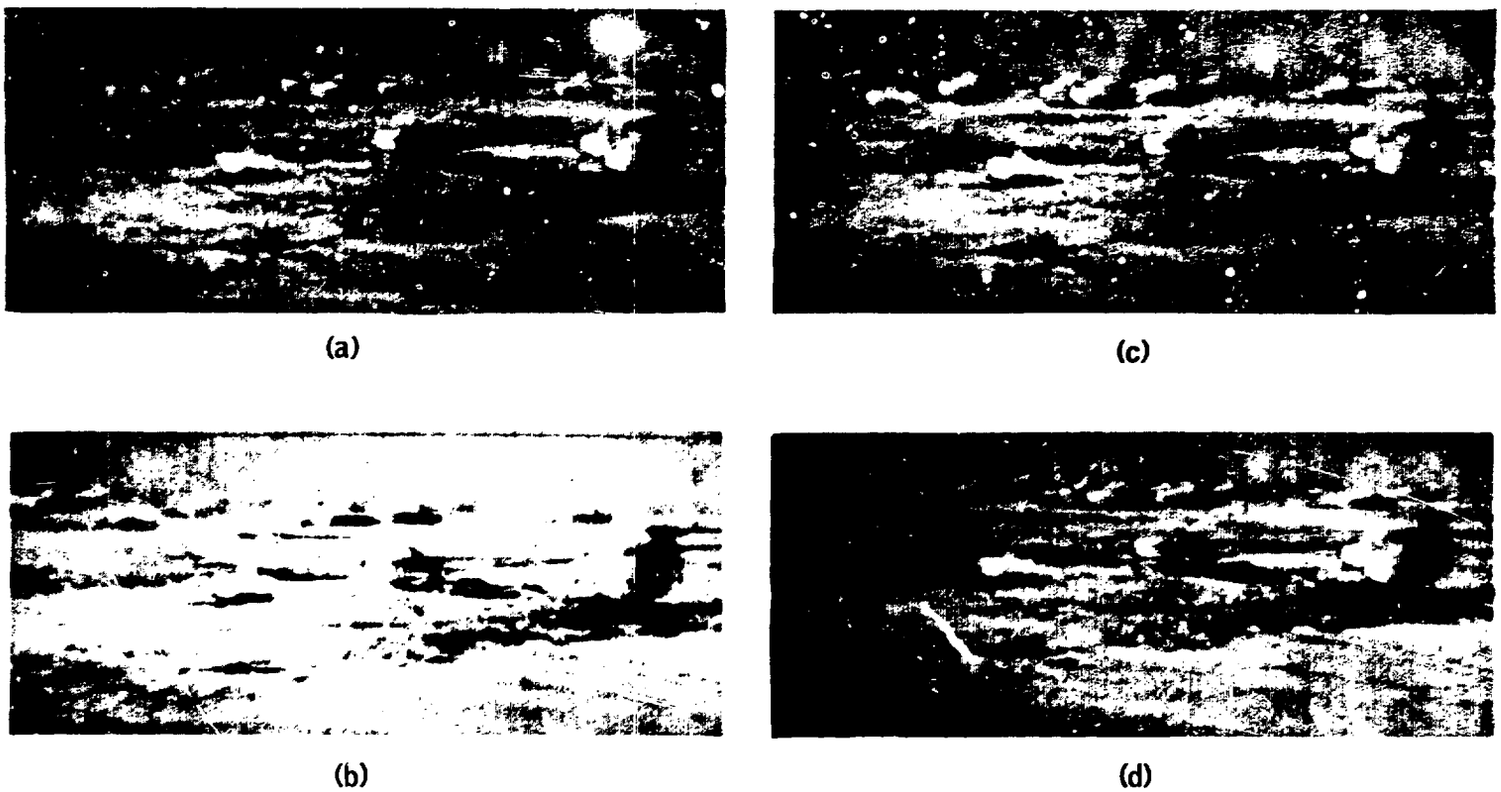
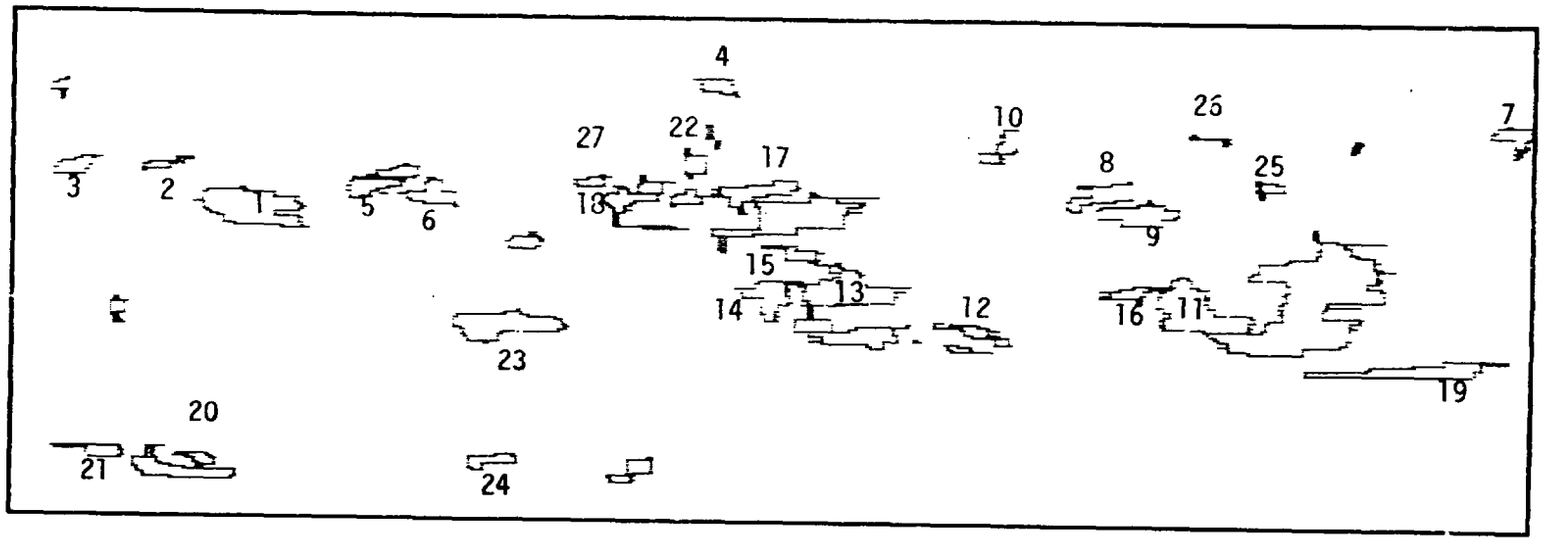
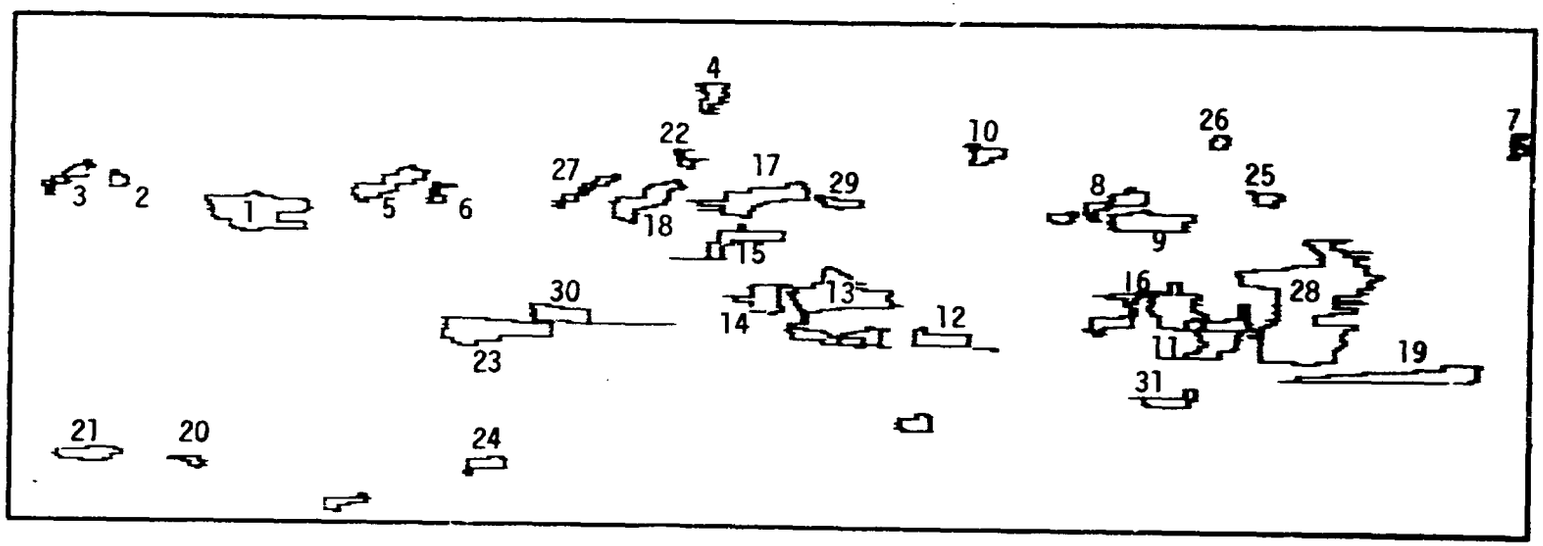


Figure 6. A Sequence of FLIR Frames

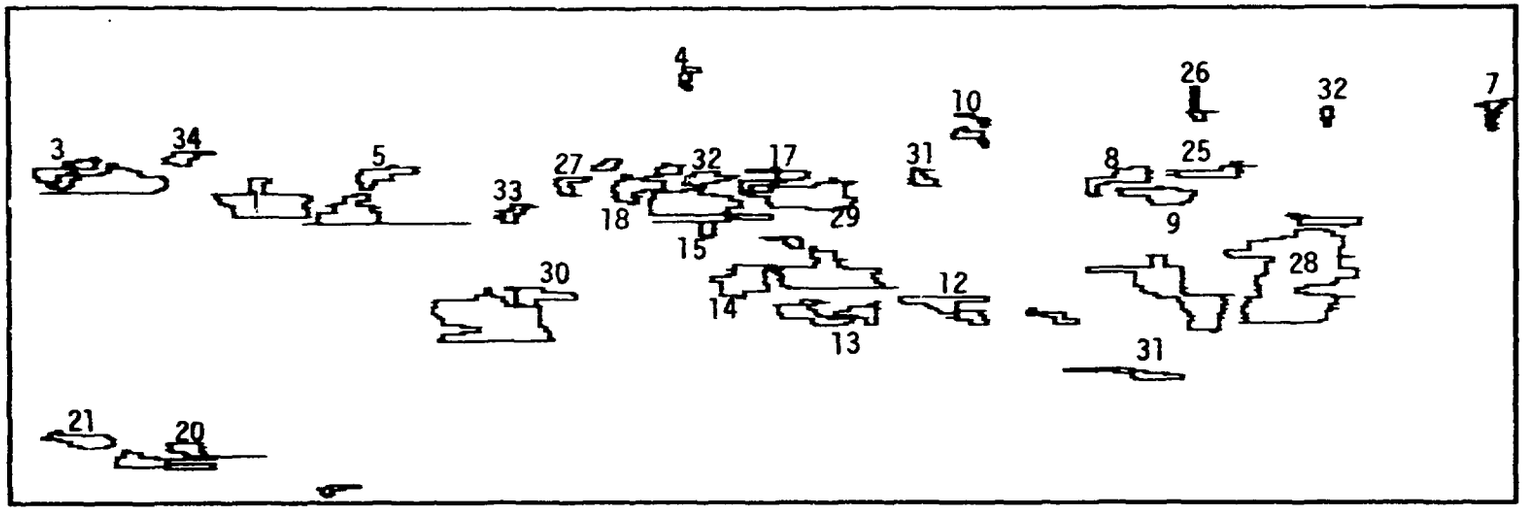


(a)

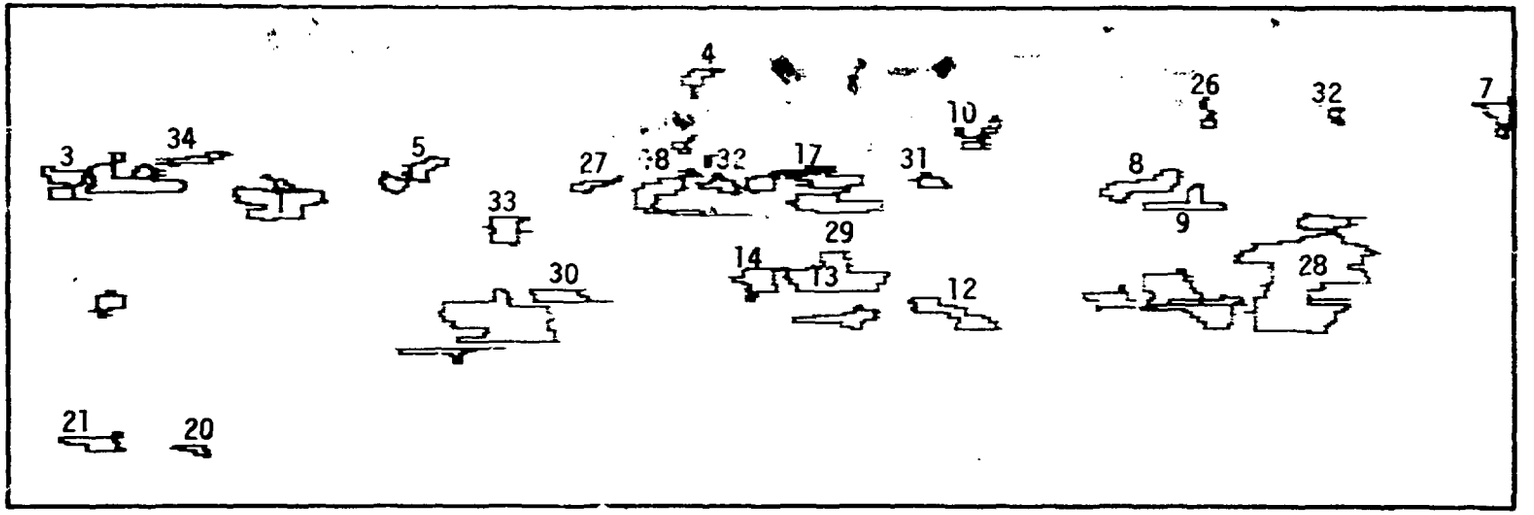


(b)

Figure 7. Results of Segmentation and Simple Object-Matching. Matched segments bear the same labels through the sequence.



(c)



(d)

Figure 7. Results of Segmentation and Simple Object-Matching--
Concluded

have been found. However, because of inconsistent segmentation and target occlusion, some object matches have been missed and some have been mis-matched.

Figure 7 points out several of the weaknesses in the simple object-matching scheme. Note that object 23 was successfully matched between the first two frames. However, in the third frame, the cold region beneath the tank was segmented along with the target. This increased the area of the object beyond the threshold (P = 0.25), which prohibited matching.

Furthermore, this method does not yield the precise motion of an object between frames. It produces matching pairs of objects and their corresponding centroid positions. Simply subtracting the centroid positions does not yield an accurate estimate of object motion, because the centroid position will vary with occlusion and with different segmentations of the object.

FAST SILHOUETTE-MATCHING ALGORITHM

The Fast Silhouette-Matching Algorithm (FSMA) achieves rapid and precise matching of objects in two frames in the presence of occlusion and inconsistent segmentations and overcomes the limitations of the simple feature-based approach. To accurately track moving objects and estimate their velocity, the movement of an object between frames must be precisely determined. This requires knowing the object motion to a pixel, or less. Furthermore, the matching algorithm must function even when the target is occluded or missegmented, by matching portions of the target which are consistent between frames.

As in the simple feature-matching algorithm, we wish to find the corresponding object(s) in the previous frame(s). However, the additional requirement is that precise position of the object must also be found. To find this movement, the outlines of the objects are aligned by the algorithm, so that those edges which have been found in both frames (that is, the consistent edges) match exactly. The displacement required for this alignment is the interframe object motion which is desired. Since the matching is done using only those edges (or portions thereof) which have been extracted in both frames, the algorithm will succeed even when the segmentation of the objects changes because of occlusion. The following paragraphs describe the algorithm.

The FSMA also uses the output of the PATS segmentation for object matching. In addition to using the object centroid position and contrast, FSMA also uses the object outline (silhouette) in the matching process.

As in the simple feature-matching algorithm, the object centroids are compared to limit the size of the search region in the current frame. The centroids are also compared to the object outlines to see if an object in the current frame could be included in the object from the previous frame.

Figure 8 illustrates this initial pruning step. If only the distance between centroids was examined, then the object in the current frame would have been incorrectly excluded from matching with the object in the previous frame. However, when the check for inclusion is made, the current object passes the test and the matching will continue. Similarly, if the centroid of an object in the previous frame falls within the outline of an object in the current frame, then the matching will continue.

This allows for precise matching in the presence of incomplete segmentation. The contrast feature is used, as in the simple feature-matching algorithm, to prevent further processing of hot-to-cold object matches. Similarly, only the sign of the contrast is checked, and the object eliminated if it does not match the object in the previous frame.

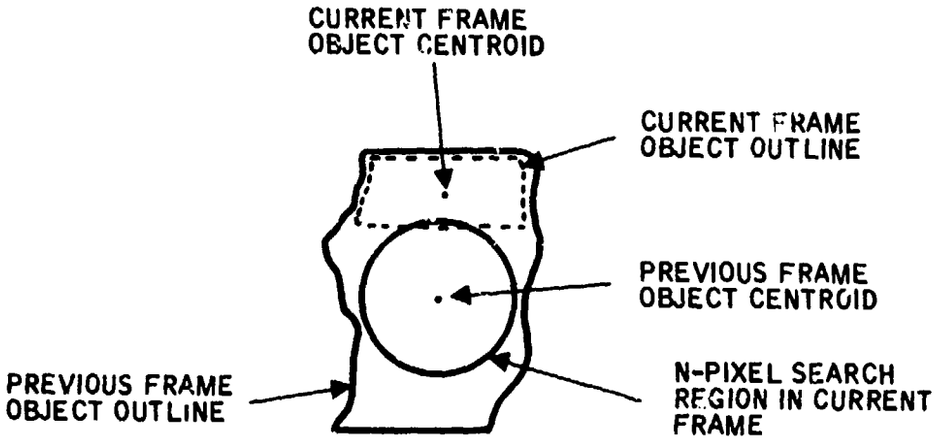


Figure 8. Centroid Test for FSMA

The precise matching is performed by the silhouette-matching algorithm. The algorithm will shift the object outline found in the previous frame until similar parts of the outline have been matched with an object in the current frame. The algorithm determines the amount of the shift by histogramming the differences in the endpoints of the object outlines. A flow chart of the algorithm is shown in Figure 9 and a description of the algorithm follows.

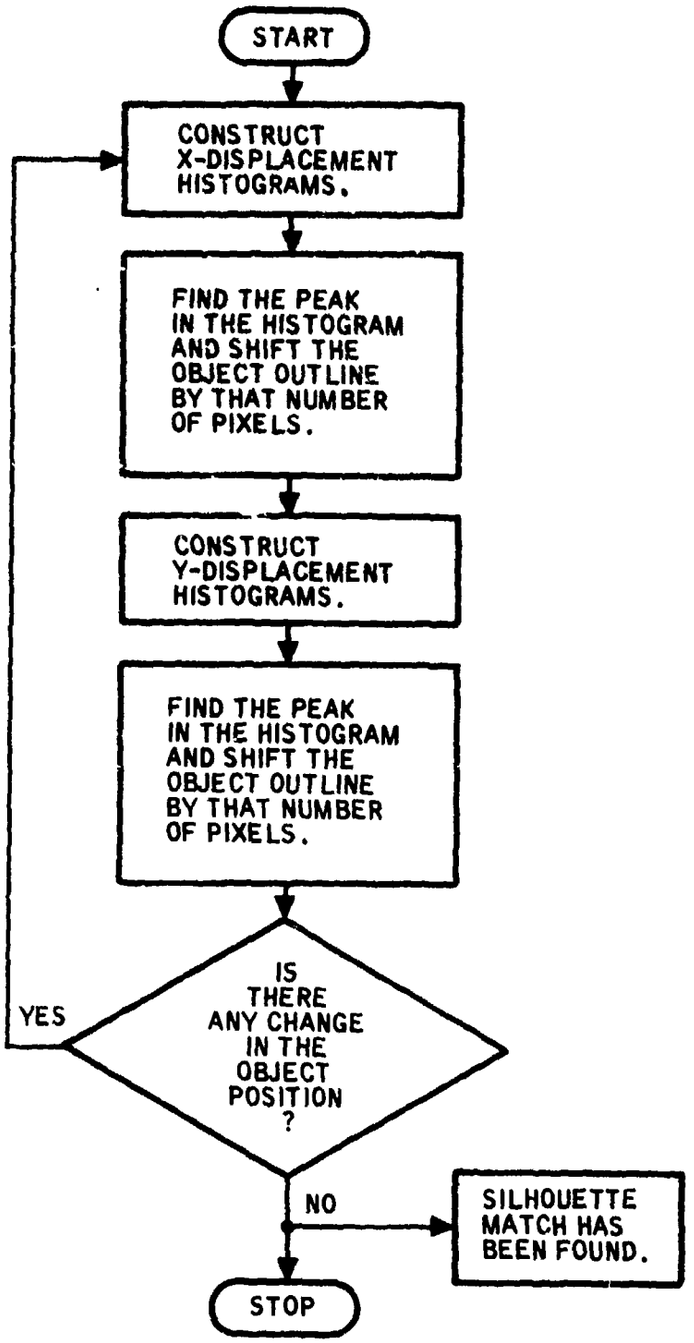


Figure 9. FSMA Flow Chart

The first step in the matching process is to calculate the left-edge X-displacement histogram. This is found by histogramming the differences in the columns positions of left-edge endpoints in each line for the two objects. The number of points in the histogram will be equal to the number of lines (rows) which contain both objects. This process is shown in Figure 10a. A similar histogram can be constructed for the right-edge endpoints of the object as shown in Figure 10b.

After forming the histograms, the X-displacement of the object is determined. The peaks in both the left- and right-edge X-displacement histograms are found. The larger of the two peaks determines the correct X-displacement. In Figures 10a and 10b, the left-edge histogram has yielded the highest peak. Therefore, the X-displacement of the object is found to be +3 pixels. Furthermore, because the right-edge histogram did not yield a peak at +3 pixels, only a left-edge match will be declared at this time.

Before forming the Y-displacement histograms, the X-displacement found in the previous step is used to displace the coordinates of the silhouette. Now, top-edge and bottom-edge Y-displacement histograms are formed as shown in Figures 10c and 10d. The peaks in these two histograms both occur at the same place, +2. In this case, the Y-displacement is set to +2, and both a top-edge and bottom-edge match are declared.

The two-pixel Y-displacement is removed from the coordinates of the silhouette, and X-displacement histograms are formed again in Figures 10e and 10f. The peaks in both the left- and right-edge histograms are equal and both occur at -2. Thus, the X-displacement is set to -2, and both a left-edge and right-edge match is declared.

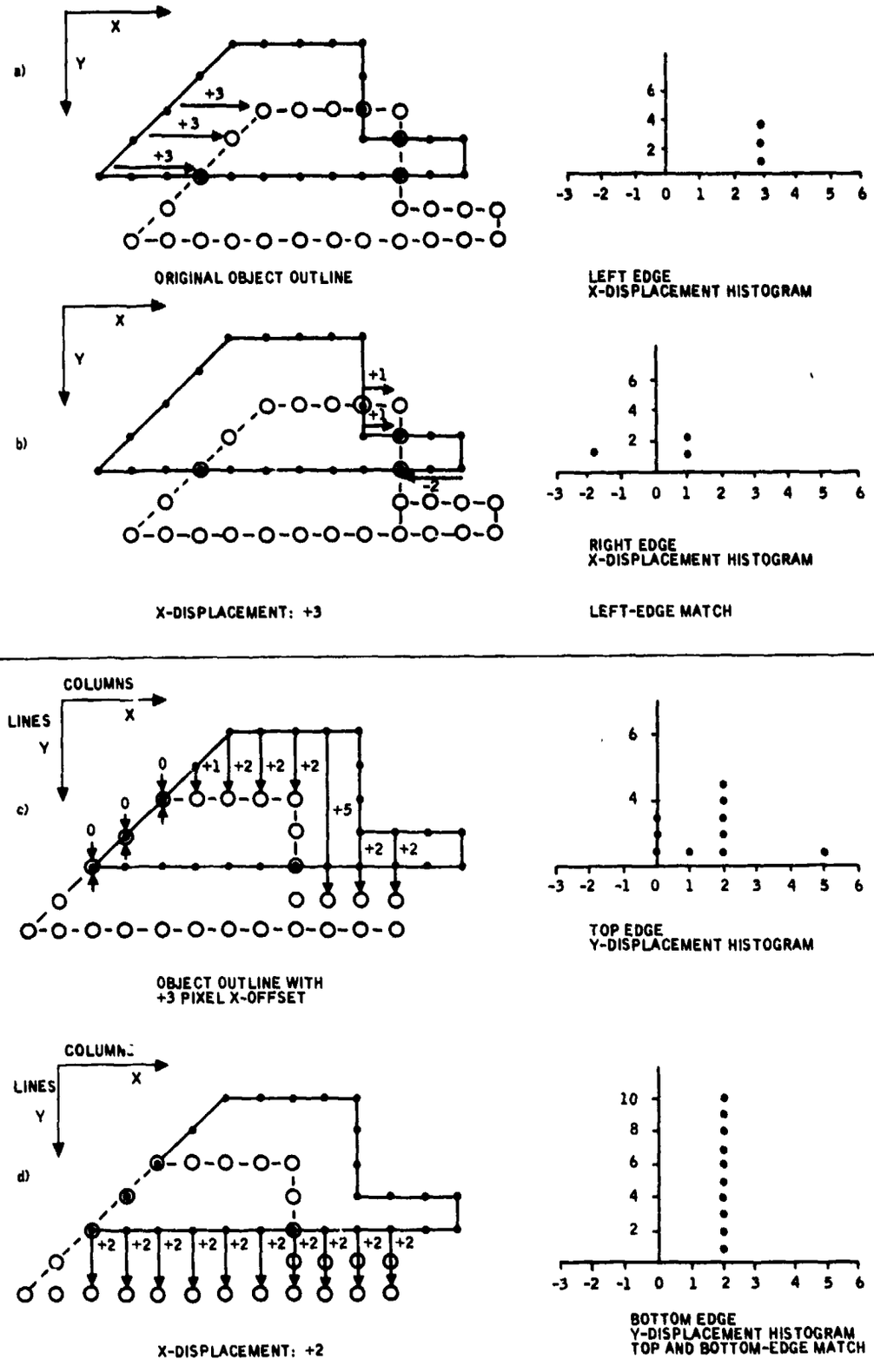


Figure 10. Silhouette-Matching Example

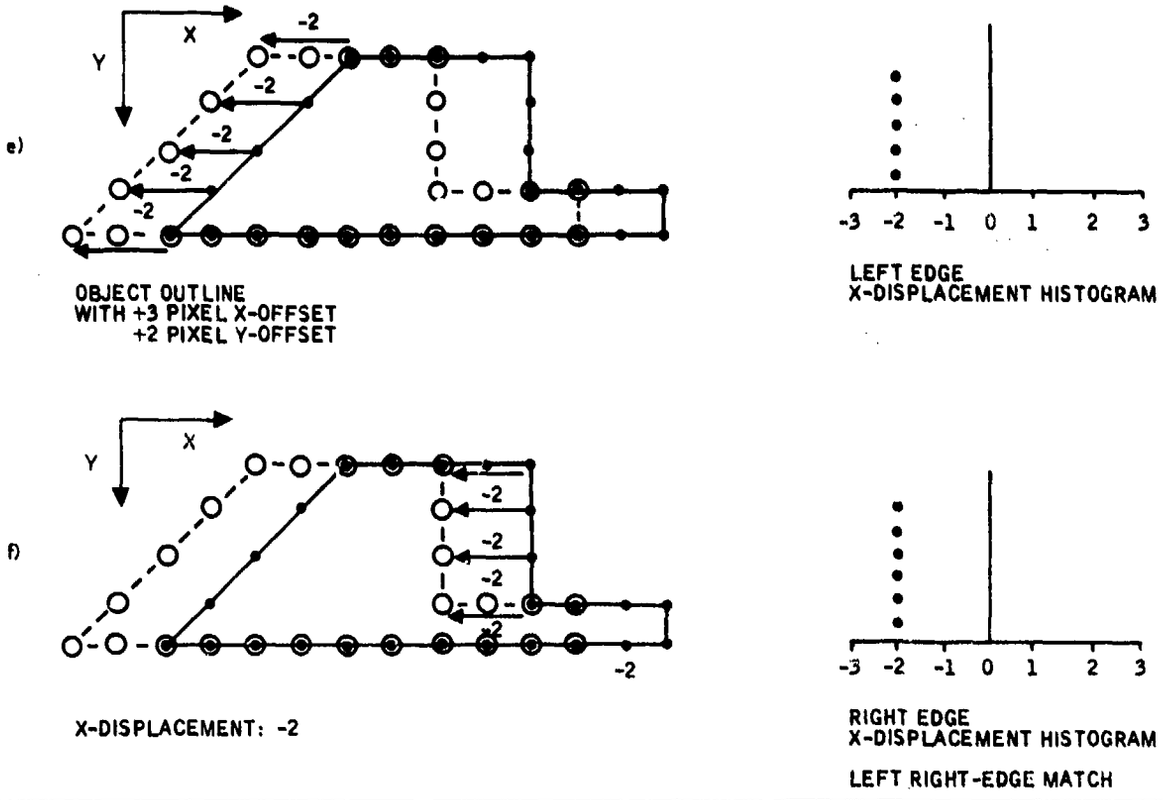


Figure 10. Silhouette-Matching Example--Concluded

The results of the matching example are shown in Figure 10g. The objects are perfectly aligned and the total object displacement has been computed. The example in Figure 10, of course, is contrived and serves to illustrate the scheme. Real-world objects will not be segmented identically in successive frames nor will they present such continuous edges.

The real-world object-matching problem will look like that of Figure 11a. Note the displacement and different segmentations in the two frames. The left- and right-edge X-displacement histograms are shown in Figure 11a. The displacement indicated by the histograms is six pixels and a left- and right-edge match was indicated. These histograms were computed in the same manner as those in Figures 10a and 10b.

Because of the nature of the PATS segmentation algorithm, the Y-displacement histograms are computed in a manner somewhat different from that which was described for Figures 10c and 10d. Because of line-wise processing implicit in PATS, objects extracted by PATS tend to have long, flat top and bottom edges. If the Y-displacement histogram were computed over the entire length of the object, it would be biased by the long top and bottom edges. Therefore, the Y-displacement histograms are computed for only those points which are in the left or right edge of the object. Figure 11b shows the original objects which the six-pixel X-adjustment and the left- and right-edge Y-displacement histograms. A displacement of -2 pixels is indicated by the histograms. Note that the peak of the right-edge histogram was used to compute the displacement (-2 pixels) although the left-edge histogram also gave two peaks of equal size at -3 and -4 pixels. This is because the histogram peaks are implicitly scaled by the total number of edge points. Thus, a peak of two out of two edge points is greater than a peak of two out of six edge points.

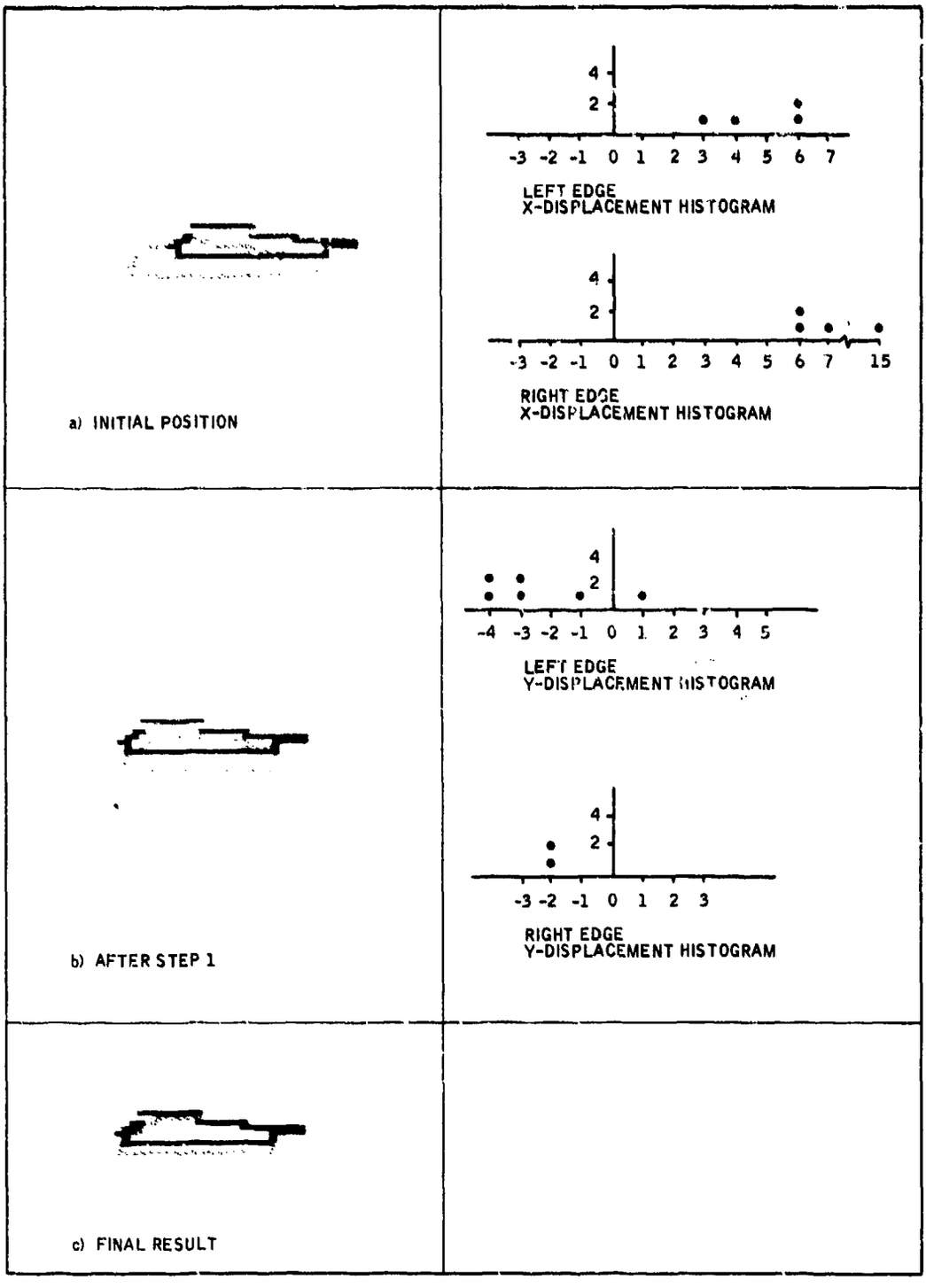


Figure 11. Real-World Silhouette-Matching: Example 1

No further displacement of the object is indicated. Note that in Figure 11c similar portions of the right edge have been aligned exactly. Other examples of Fast Silhouette-Matching are shown in Figures 12 and 13. Figure 12 is the outline of a moving tank, while Figure 13 is a group of trees.

The result of applying the FSMA to all the objects in a tactical scene is shown in Figures 14 and 15. Figure 14 shows two FLIR scenes taken approximately 240 msec apart. Figure 15 shows the segmentations of the two scenes with matching objects bearing similar labels. Note that object 4 in the first frame was segmented into two objects in the second frame. Both these objects were found to match object 4. Similarly, note that the two objects labeled "3" in the first frame have been correctly matched to one object in the second frame. These examples of one-to-many and many-to-one matching show the capability of the algorithm to find matches in the presence of inconsistent segmentations.

A significant feature of this iterative algorithm is its rapid convergence. In Figure 10, three iterations were required to find the precise silhouette match. An exhaustive search of all possible object positions would have required, in the worst case, searching an area of 4 x 4 pixels, or 16 iterations, to find the two-pixel motion of the target. Furthermore, the number of iterations required by an exhaustive search technique will be proportional to the square of the allowed target motion. The convergence of the FSMA, on the other hand, does not depend directly on the amount of displacement.

Another feature of the algorithm is that it allows the tracking of a specified location on a target. Given a point on the target in one frame, we wish to find that same point in the next frame. The FSMA tells us which edges of the object have been matched

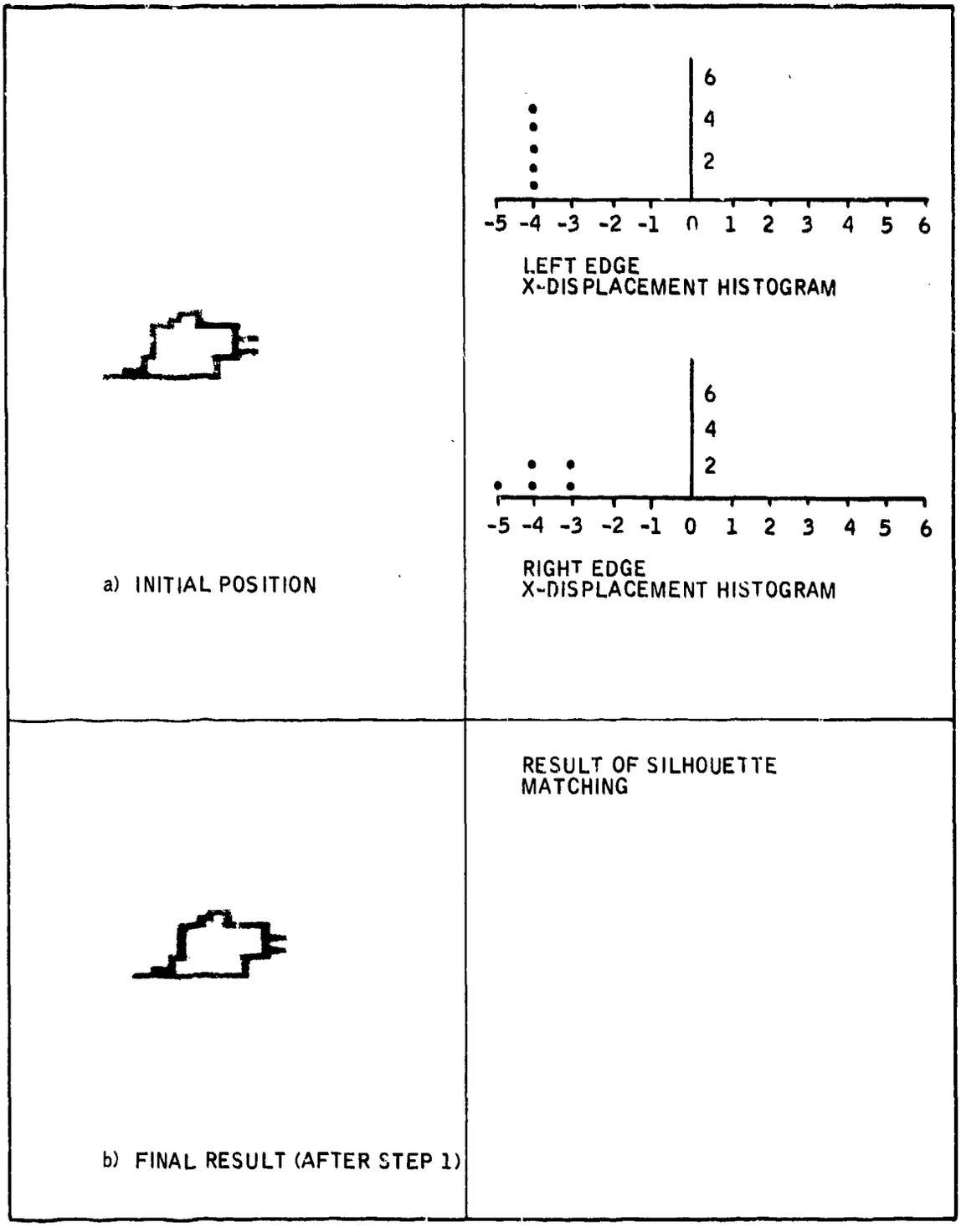


Figure 12. Real-World Silhouette-Matching: Example 2

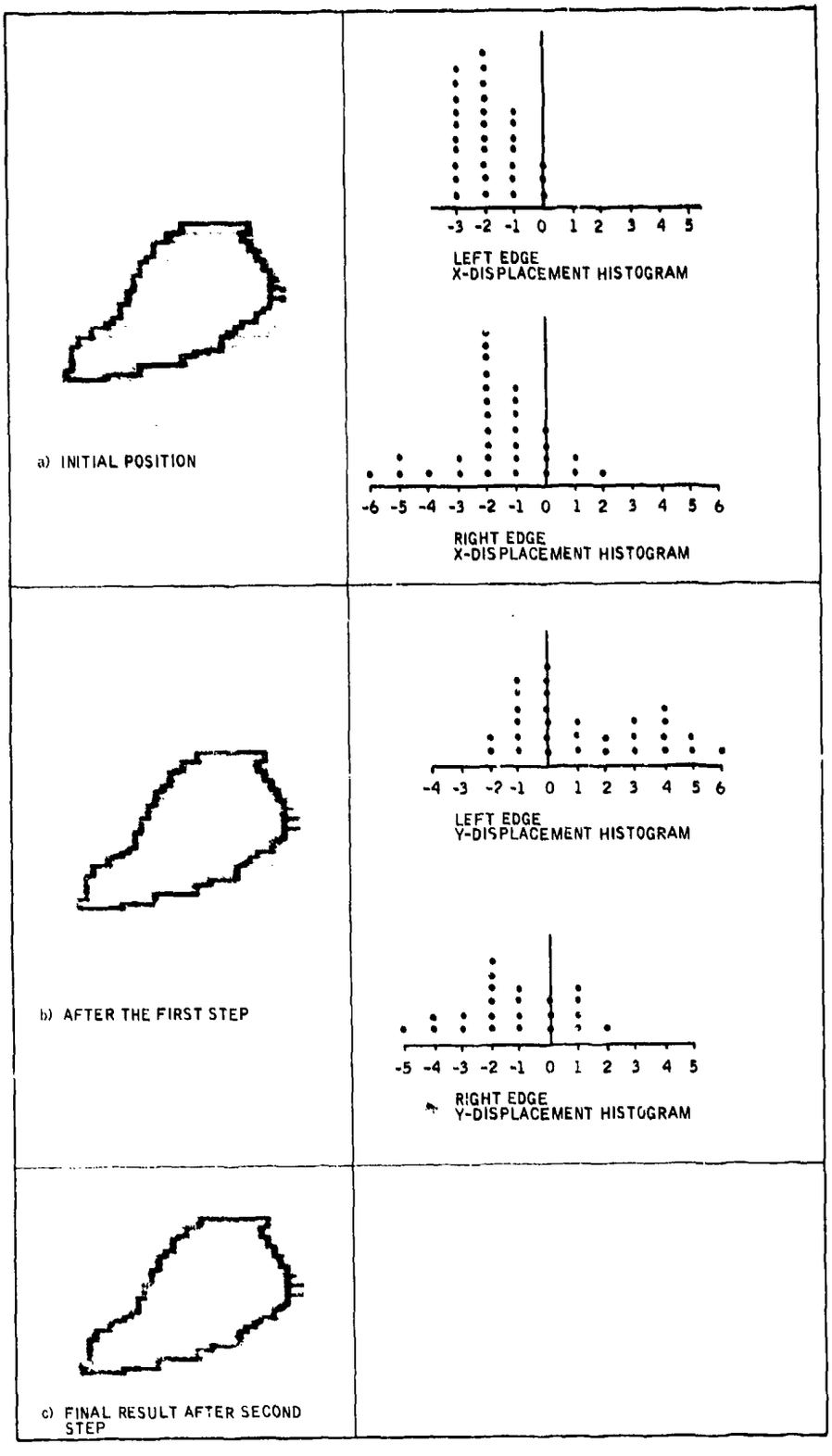
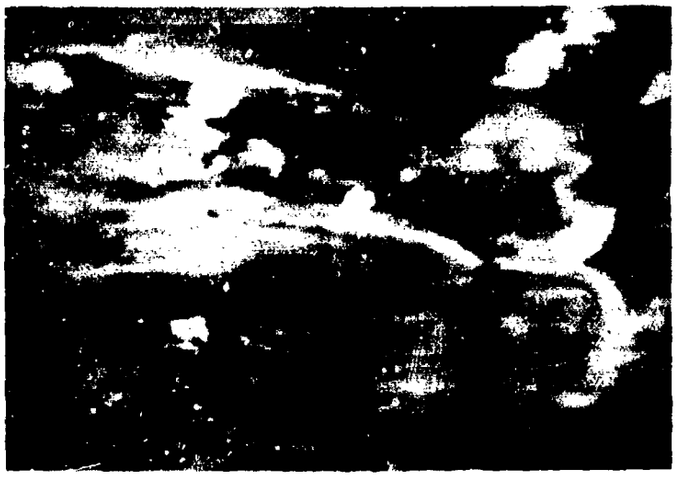


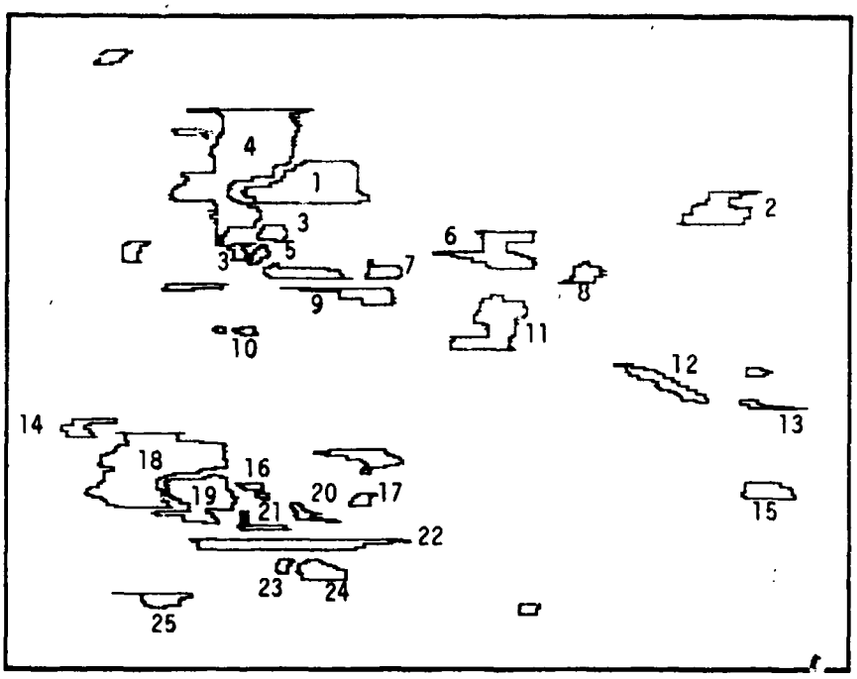
Figure 13. Real-World Silhouette-Matching: Example 3



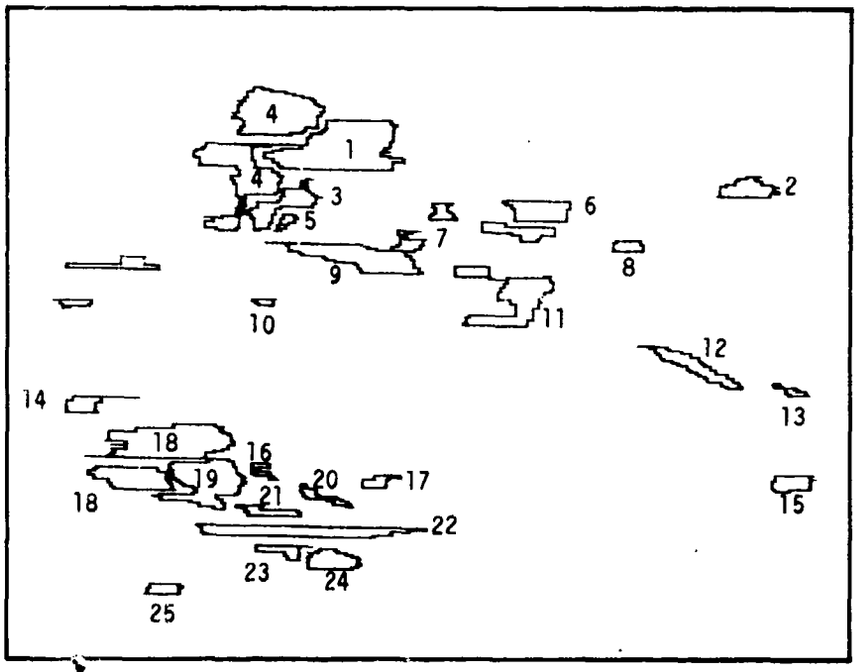
(a)

(b)

Figure 14. Two FLIR Images, From a Sequence From a Moving Platform



(a)



(b)

Figure 15. Result of Segmentation and Silhouette-Matching. Note 1-to-many match of object 4 and many-to-1 match of object 3.

between frames. If we calculate the location relative to the matched edges in the previous frame, we can then find the same location, relative to the matched edges, in the current frame. In this manner the specified location can be tracked between frames. This is critical for the homing scenario and for laser designation.

The FSMA has performed well on the test images which have been tried. Most matches are found within two to three iterations and few incorrect matches are made. However, the sensitivity of the algorithm to the initial object positions and extremely different segmentations has not been studied. The study of these topics as well as the verification of the algorithm on a larger sample of objects will be done during the next reporting period.

We have demonstrated two object matching schemes in this section. We have found that the simple object-matching scheme succeeds in finding corresponding objects in successive frames in unambiguous cases. Because of its computational simplicity, it is useful in providing an initial estimate of scene motion for input to the scene model, as we will see in the next section. It also serves to compute the initial values for the more sophisticated silhouette-based object-matching algorithm. The silhouette-based object-matching algorithm was found to perform extremely well even in the presence of target obscuration and inconsistent segmentations from one frame to the next. The algorithm was also demonstrated to be computationally elegant and simple and does not require an exhaustive search to find the optimum match. Both the simple algorithm and the sophisticated silhouette-based algorithm are used in the advanced target systems simulation, as we will see in the systems simulation section.

SECTION IV
SCENE MODEL

The primary function of the scene model is to keep track of and infer information about objects in the scene as well as the platform dynamics derived from the analysis of the previous frames. More specifically, the scene model comprises:

- Platform dynamics (position and velocity)
- Object dynamics
- Object shapes and classifications
- Occlusion prediction
- Shape prediction
- Background prediction

The platform dynamics correspond to the motion of the sensor and the RPV (or the AAH) and its impact upon the received image. Knowledge of the platform dynamics is useful both in finding the relative motion of targets with respect to the scene and in providing scene-track information to the platform gimbals if scene stabilization is required. Platform dynamics are computed from the positions of corresponding clutter (stationary) object matches.

Individual object positions computed by the segmentor and the object matcher are used to compute the individual object dynamics over several frames. Object dynamics can be represented either relative to the sensor field of view or relative to the scene after the platform dynamics have been accounted for. The former

is useful for multitarget tracking -- say for laser designation -- where only the positions of the target relative to the current field of view are desired. The latter also estimates the motion of the target relative to the scene (independent of the sensor motion) and permits target/clutter discrimination based on motion.

Because the scene model keeps track of all the object positions, as well as the background characteristics in different regions of the image, it can be used to predict the occlusion of objects that are moving toward each other, an object which moves into a low-contrast background, etc. The shapes of occluded objects can also be predicted so that the object matcher can use the predicted shapes to perform better matches in successive frames. Furthermore, the artificial intelligence capability of the scene model will allow inference of the target shape from its segmentations in previous frames. For example, if multiple segments of an object appear to move together over several frames, then the inference is that they belong to the same object.

In this reporting period, we concentrated primarily on the estimation of platform displacement from the result of object-matching algorithms described in the previous section. We have successfully demonstrated that the platform dynamics can be computed accurately (to the pixel) using the techniques described below.

Three increasingly complex models of scene motion have been derived. The three-parameter model estimates rotation and translation of the field of view from one frame to the next. Therefore, it does not account for the motion of the sensor in space. A more complex five-parameter model allows sensor motion, but only in the vertical plane containing the target. A complete six-parameter model can account for sensor translation and rotation as well as

the sensor motion in free space in all three degrees of freedom. This model has been implemented in the system simulation and the results are shown in the system simulation section.

THREE-PARAMETER ESTIMATION OF SCENE MOTION

Consider a sensor fixed at a point in space and free to rotate about all three of its axes as shown in Figure 16. We will show the effects of these three rotations on the field of view (FOV) of the sensor.

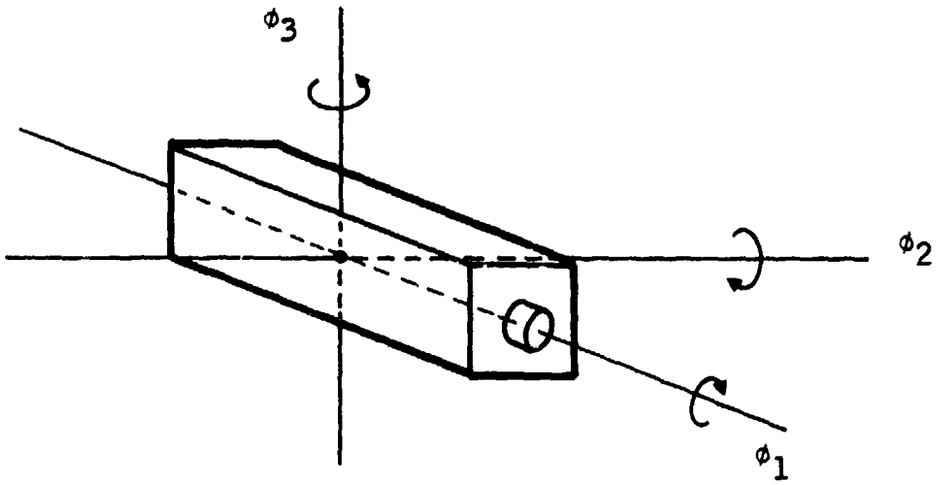


Figure 16. A Sensor Fixed in Space

Rotation about the ϕ_1 axis by an angle, θ, will produce a similar rotation of the FOV. The rotation of the sensor has caused an apparent rotation of all objects in the FOV. This is illustrated in Figure 17. If the sensor is rotated about the ϕ_2 axis, as shown in Figure 18, then there will be an apparent vertical motion of

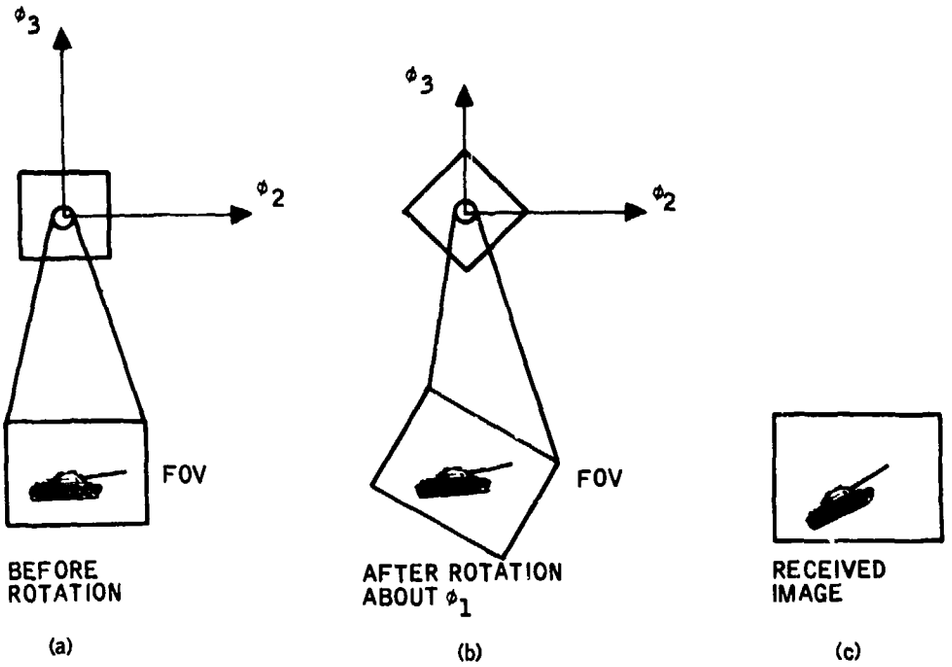


Figure 17. Effect of Rotation About ϕ_1

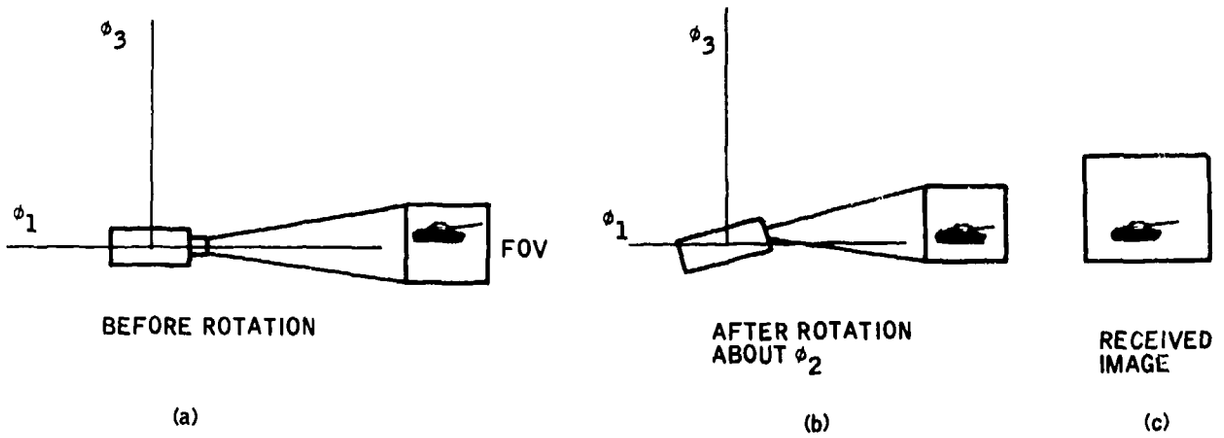


Figure 18. Effect of Rotation About ϕ_2

all the objects in the FOV. The slight distortion caused by the different viewing angle is neglected by this model. Similarly, rotation about the ϕ_3 axis will cause an apparent horizontal displacement of all the objects in the FOV.

If the platform velocity is small, compared with the sensor instability, this three-parameter model estimates the apparent motion of the sensor. This is done by finding least squares estimates of the following quantities:

- θ - angle of rotation about ϕ_1.
- X_o - horizontal displacement caused by rotation about ϕ_3.
- Y_o - vertical displacement caused by rotation about ϕ_2.

The locations of matched-object pairs are used as input to the least-square estimator as follows.

Let (x,y) be the position of an object in the previous frame and (x´,y´) be the position of the matching object in the current frame.

Find θ, x_o, y_o such that

$$E\left[\left\|\begin{bmatrix}\cos\theta & \sin\theta & x_o\\ -\sin\theta & \cos\theta & y_o\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix} - \begin{bmatrix}x'\\ y'\end{bmatrix}\right\|^2\right] = \mathcal{E}$$

is a minimum, or

$$E\Big[(x \cos\theta + y \sin\theta + x_o - x')^2 + (-x \sin\theta + y \cos\theta + y_o - y')^2\Big] = \mathcal{E}$$

is a minimum.

Differentiating with respect to θ, x_o, y_o and distributing the expectation yields the following equations:

$$\frac{\partial \mathcal{E}}{\partial \theta} = (Ey)\, x_o \cos\theta - (Ex)\, x_o \sin\theta - (Ex)\, y_o \cos\theta - (Ey)\, y_o \sin\theta + \Big[(Exy') - (Eyx')\Big] \cos\theta + \Big[(Exx') + (Eyy')\Big] \sin\theta$$

$$\frac{\partial \mathcal{E}}{\partial x_o} = (Ex') - (Ex) \cos\theta - (Ey) \sin\theta - x_o$$

$$\frac{\partial \mathcal{E}}{\partial y_o} = (Ey') + (Ex) \sin\theta - (Ey) \cos\theta - y_o$$

Equating these to zero and solving for θ, x_o, and y_o yields the following equations:

$$\tan\hat{\theta} = \frac{S_{yx'} - S_{xy'}}{S_{xx'} + S_{yy'}} \qquad (1)$$

$$\hat{x}_o = (Ex') - (Ex) \cos\theta - (Ey) \sin\theta \qquad (2)$$

$$\hat{y}_o = (Ey') + (Ex) \sin\theta - (Ey) \cos\theta \qquad (3)$$

where

$$S_{ab} = (Eab) - (Ea)\,(Eb)$$

Because we are given the pairs of points (x,y) and (x´,y´), we can calculate Syx´, Sxy´, Sxx´, and Syy´. Combining these in Equation (1) yields tan θ. Since θ represents the rotation of the sensor between frames, we can assume that θ lies in the interval $-90^{\circ} \leq \theta \leq 90^{\circ}$ and take the inverse tangent to find θ. Then, using Equations (2) and (3) we can find the values of x_o and y_o.

FIVE-PARAMETER ESTIMATION OF SCENE MOTION

The three-parameter solution assumes that the motion of the sensor through space can be neglected. For high-velocity aircraft (munition or RPV), this is not a valid assumption. The five-parameter model allows the sensor to rotate about its three axes and also to move in the plane defined by the ϕ_3 and ϕ_1 axes. Sensor motion within this plane causes the FOV to increase (as the sensor moves away from the scene) or decrease (as the sensor approaches the scene). This change in the FOV introduces a linear scale change, in both horizontal and vertical directions, into the transformation from the last frame to the current frame. The transformation then assumes the following form:

$$\begin{bmatrix} (\cos\theta + k_1)\sin\theta & x_o \\ -\sin\theta & (\cos\theta + k_2)\,y_o \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

Note here that k_1 and k_2 are scale factors in the x and y directions, respectively.

The least-square estimates of the 5 parameters θ, x_o, y_o, k_1, and k_2 can be calculated and yield the following values:

$$\sin\theta = \frac{S_{xy}S_{yy'}S_{xx} - S_{xx}S_{yy}(S_{xy'} - S_{yx'}) - S_{xy}S_{xx'}S_{yy}}{(S_{xx}S_{yy} - S_{xy}S_{xy})(S_{xx} + S_{yy})}$$

$$k_1 = \frac{S_{xx'} - S_{xx}\cos\theta - S_{xy}\sin\theta}{S_{xx}}$$

$$k_2 = \frac{S_{yy'} - S_{yy}\cos\theta + S_{xy}\sin\theta}{S_{yy}}$$

$$x_o = Ex' - Ex(\cos\theta + k_1) - Ey\sin\theta$$

$$y_o = Ey' + Ex\sin\theta - Ey(\cos\theta + k_2)$$

SIX-PARAMETER ESTIMATION OF SCENE MOTION

The six-parameter scene model allows the sensor to rotate about its axis and also move in any direction in space. This is the transformation which is currently used in the system simulation. It has the following form:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

The least square estimates for the a_{ij} will minimize

$$E\left[(a_{11}x + a_{12}y + a_{13} - x')^2 + (a_{21}x + a_{22}y + a_{23} - y')^2\right]$$

Solving for a_{ij} yields the following two matrix equations:

$$\begin{bmatrix} Ex^2 & Exy & Ex \\ Exy & Ey^2 & Ey \\ Ex & Ey & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \end{bmatrix} = \begin{bmatrix} Exx' \\ Eyx' \\ Ex' \end{bmatrix} \quad \text{and}$$

$$\begin{bmatrix} Ex^2 & Exy & Ex \\ Exy & Ey^2 & Ey \\ Ex & Ey & 1 \end{bmatrix} \begin{bmatrix} a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} Exy' \\ Eyy' \\ Ey' \end{bmatrix}$$

Given the positions of the matched objects (x,y) and (x',y'), the expected values can be calculated and the two sets of equations can be solved for the six parameters.

SECTION V
SYSTEM SIMULATION

The previous sections discuss the progress during this reporting period on two key facets of the advanced target tracker system -- the object-matching techniques and the scene model. These individual techniques have been incorporated into a complete systems simulation of the advanced target-tracker system in the Honeywell Image Processing Facility. This simulation allows the evaluation of the algorithms as they are developed in the system context. This section discusses the status of the system simulation and simulation results on two sequences of FLIR images from moving and stationary platforms. The results demonstrate precise tracking capability with multiple targets and high clutter scenes, as well as moving-target detection capability, even with unstabilized moving platforms. This system simulation will be expanded as new algorithms and software are developed for such factors as occlusion prediction, target/background signature prediction, and advanced scene models.

A block diagram of the current system simulation is shown in Figure 19. The simulation currently consists of the following software modules:

- PATS segmentation
- Simple object-matching
- Fast silhouette-matching

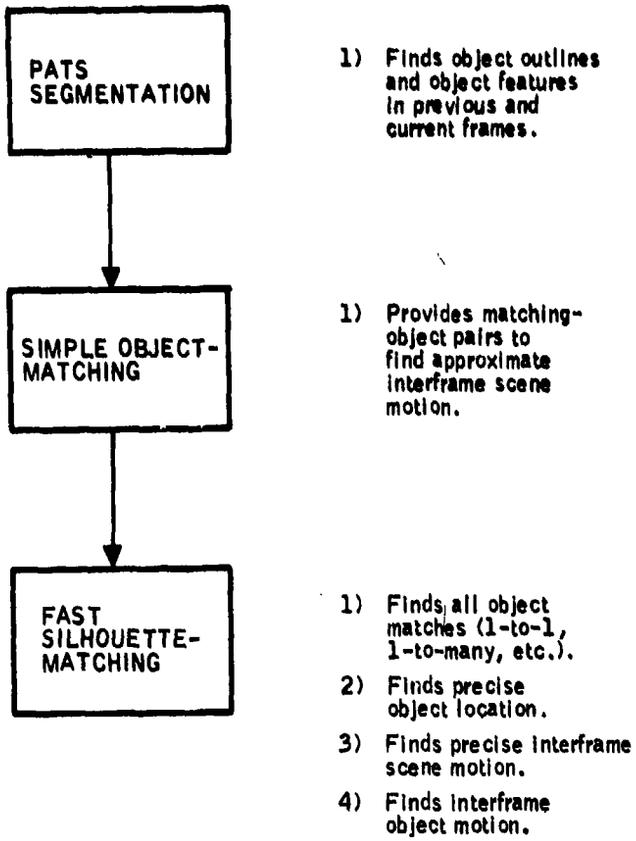


Figure 19. System Simulation Block Diagram

In the system simulation, the PATS segmentation is applied to the two input frames. This produces a list of object outlines and features which will be matched. The simple object-matching algorithm matches objects between the two frames to find the approximate interframe scene motion. This approximate transformation is applied to the objects in the previous frame and the Fast Silhouette-Matching Algorithm is applied. The FSMA will match all the objects which are present in both scenes and find their exact displacement. Using these matches and the determined displacements, a finer estimate of the scene motion can be computed. Finally, the results of scene motion model and object matching can be combined to yield an estimate of the interframe object motion.

The advanced tracker system simulation block diagram has been expanded in Figure 20. We can see that simple object-matching is first applied to the objects found by the PATS segmentations. The centroids of the matched objects are used to calculate an approximate transformation from the previous frame to the current frame. This transformation is then applied to the objects from the previous frame. Simple object-matching is applied to the adjusted objects from the previous frame and the current frame objects. The second application of the simple object-matching scheme will, in general, find more matches than the first. This is due to the better frame alignment after applying the approximate transformation.

If more matches have been found during the second pass, then a new transformation will be computed. This sequence is iterated until no new matches can be found by the simple object-matching scheme.

After simple object-matching, the two frames have been brought into approximate alignment and the Fast Silhouette-Matching Algorithm is applied. The FSMA will find all object matches between the two frames including the one-to-one, one-to-many, many-to-one, and many-to-many object matches which were not found by the simple matching process. The FSMA also determines the precise location of the objects in the current frame. Using these precise locations, an accurate estimate of the interframe transformation can be made.

Using this transformation, we can predict the location of a previous frame object in the current frame. Subtracting this predicted location from the actual location found by the FSMA yields an estimate of target motion relative to the background. This velocity will be used in future versions to predict occlusion and to aid in tracking the target if it leaves the FOV.

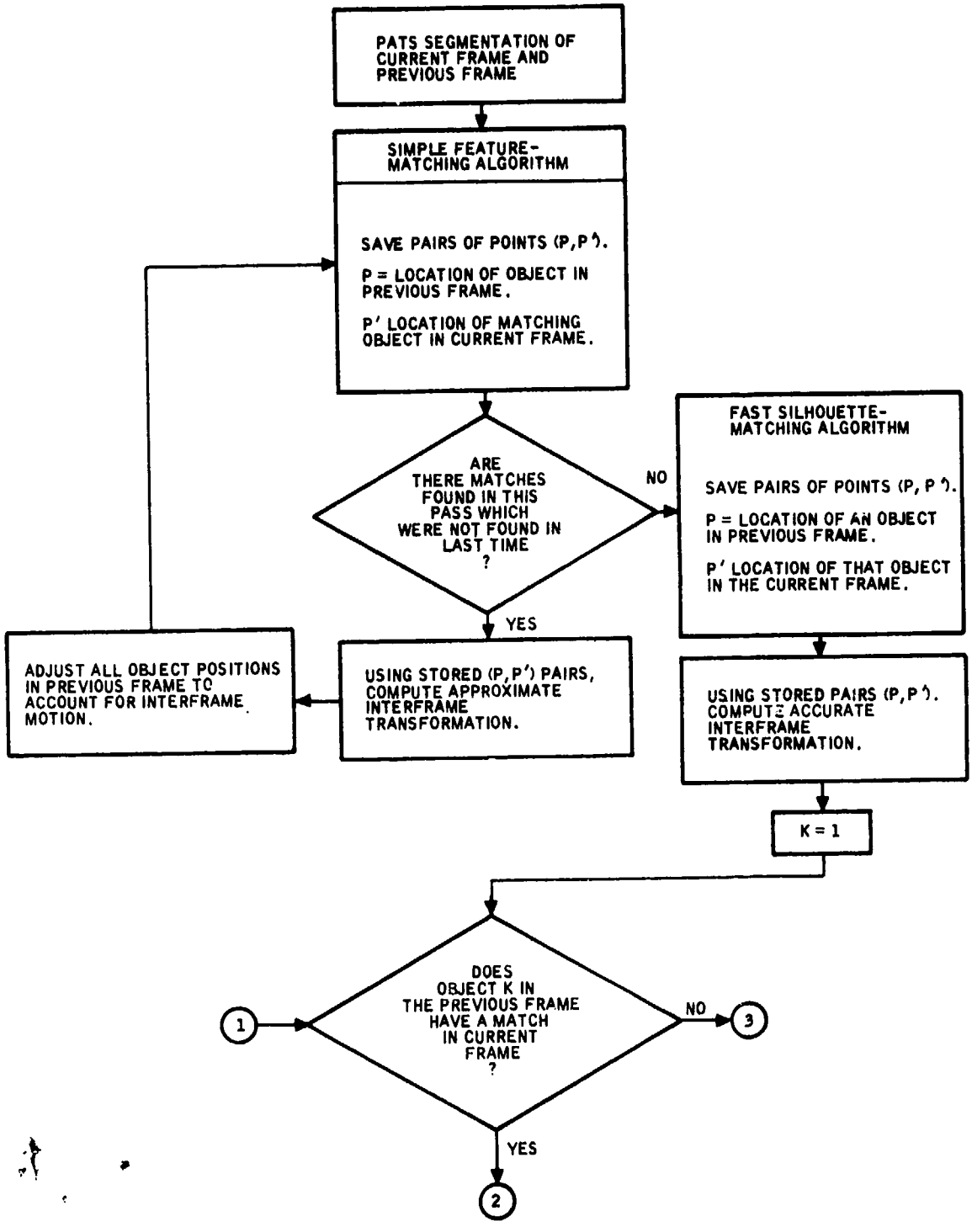


Figure 20. System Simulation Flow Chart

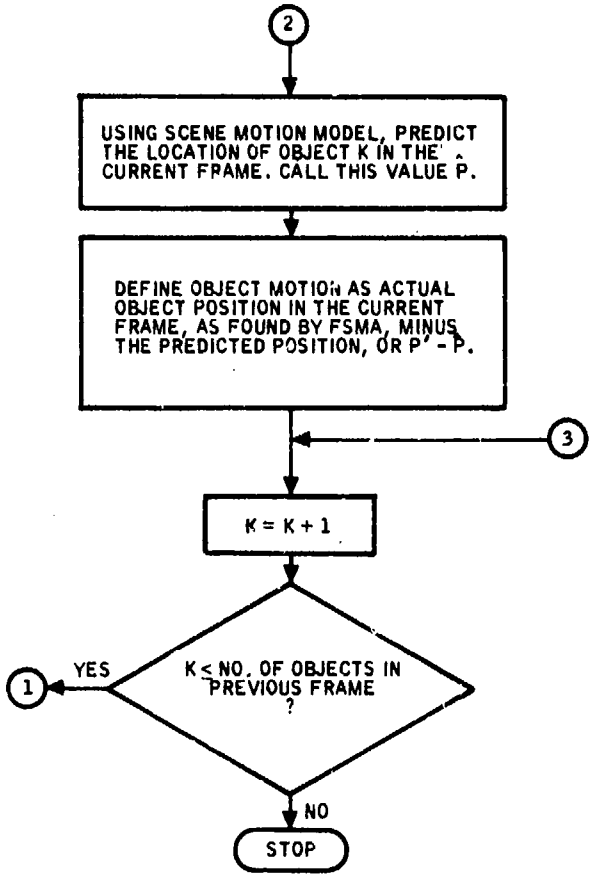
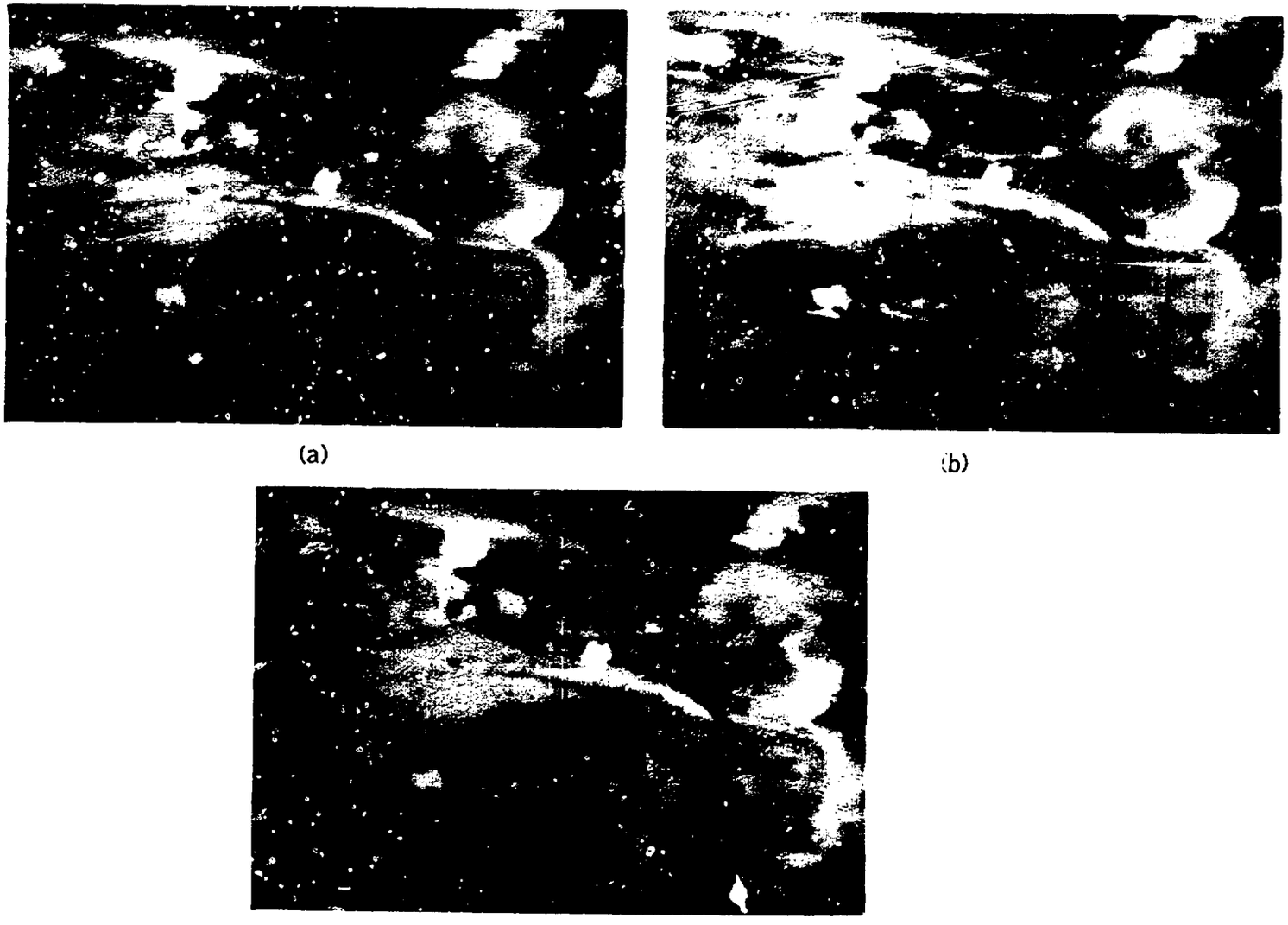


Figure 20. System Simulation Flow Chart--Concluded

Results of the system simulation are shown in Figures 21 through 30. Figure 21 shows a sequence of three FLIR images from high velocity aircraft. A small moving target can be observed near the center of the image. Because of the sensor motion, we can see the translation and rotation between the images. Furthermore, the movement of the aircraft toward the target has caused dilation (scale change) between frames.

Figure 22 shows the three segmentations superimposed. Again, note the translation and different segmentations between frames. In Figure 23 the frames have been aligned to account for sensor



(a)

(b)

(c)

Figure 21. Three FLIR Images

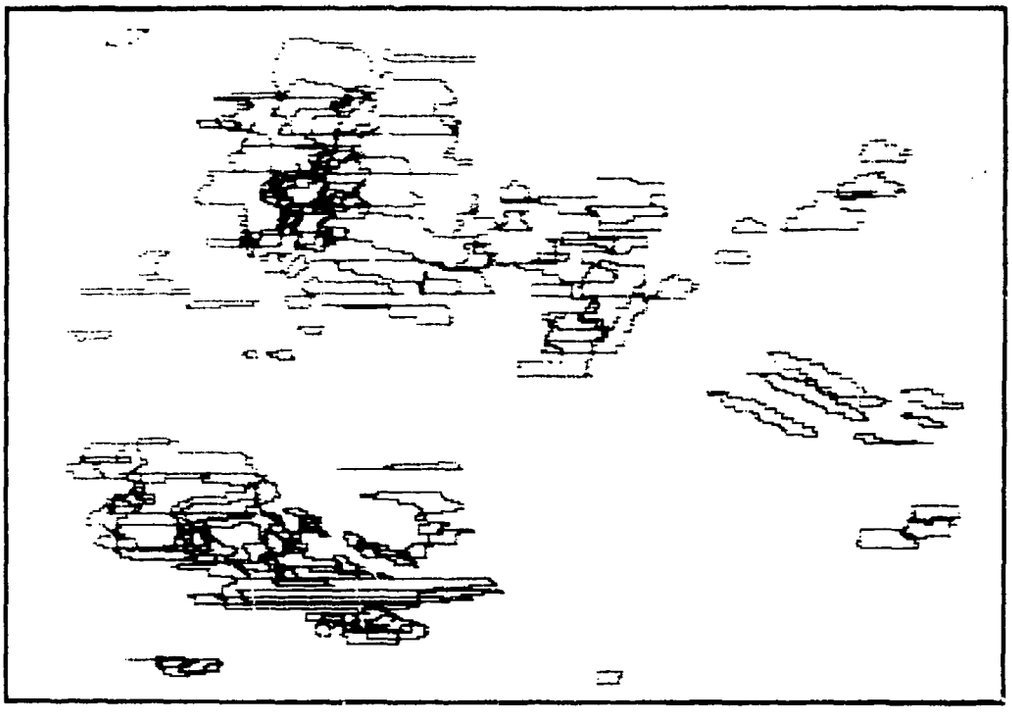


Figure 22. Unaligned Object Outlines From Three Frames Shown Superimposed

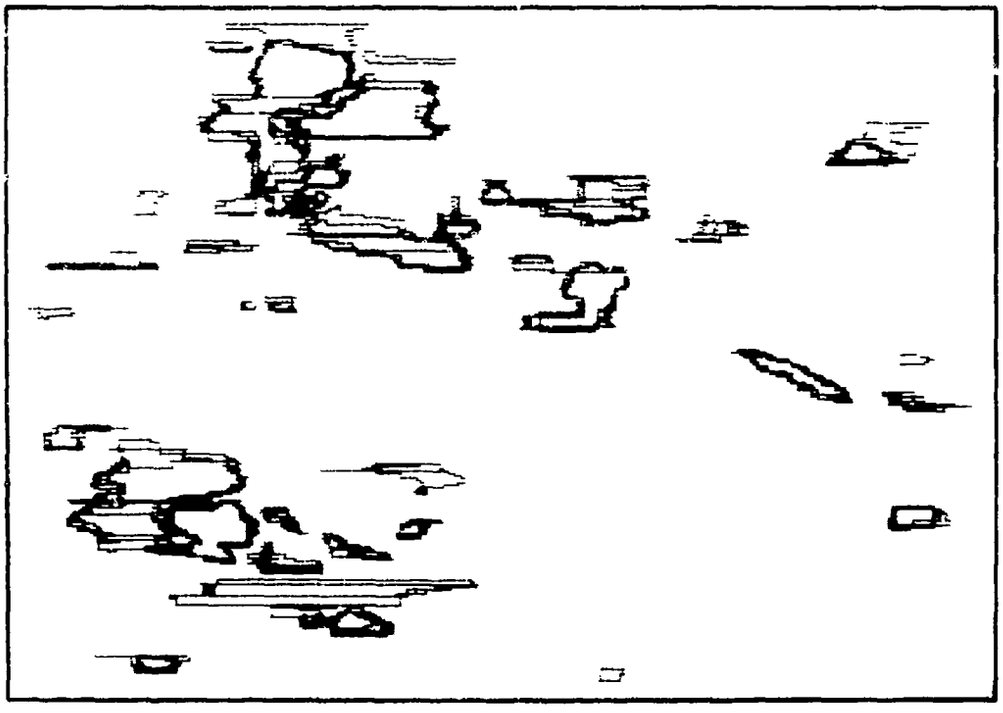


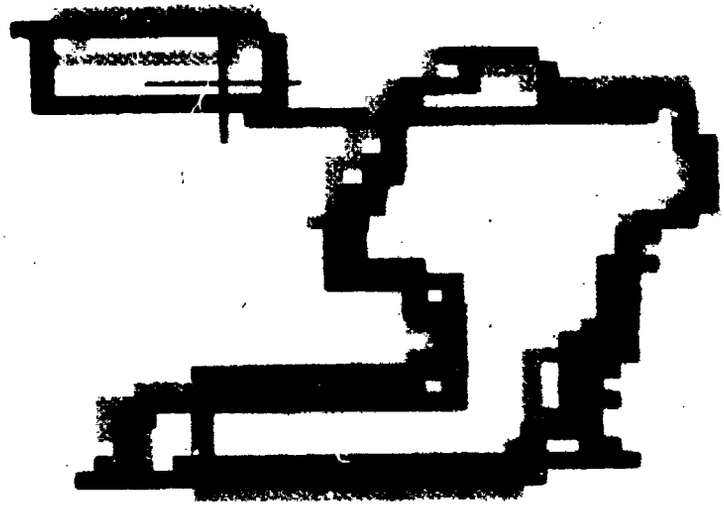
Figure 23. Object Outlines After Alignment

motion. Note the alignment of the stationary clutter objects and the movement of the tank between frames in the magnified portions of the aligned frames in Figures 24 and 25.

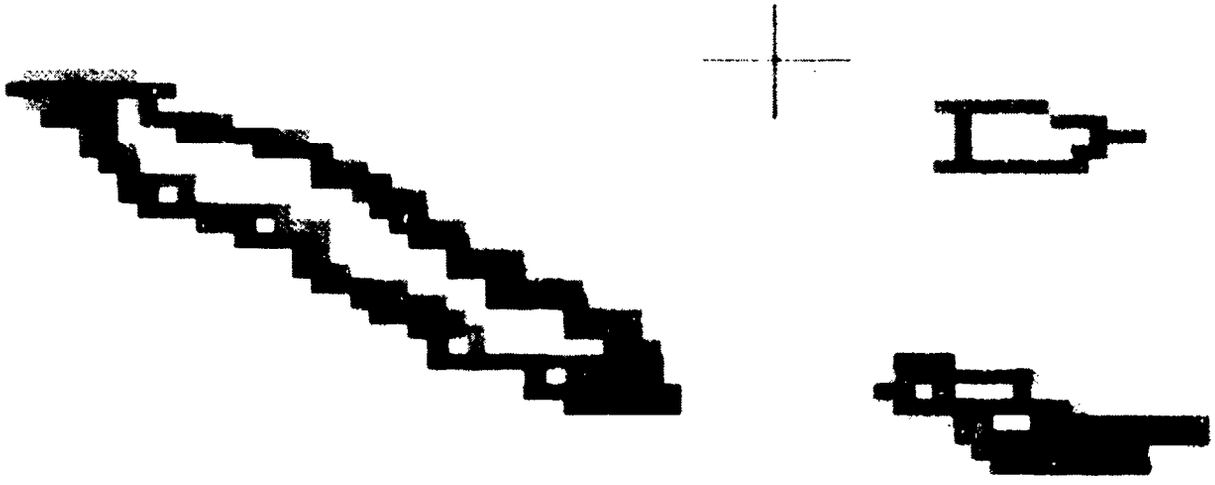
To judge the effectiveness of this scheme for moving target detection, the apparent motion of each object (which appeared in all three frames) was computed after compensating for the sensor motion. The moving tank had a cumulative displacement of seven pixels (over the three-frame sequence), while all other objects gave rise to net displacements of less than two pixels. Note that this encouraging result was obtained from only three frames. It is expected that filtering the displacement over several consecutive frames with Kalman filters will discriminate the consistent target motion even better.

Figure 26 shows two FLIR frames from a sequence of 10 which were input to the system simulation. Even though these frames were taken from a ground platform, they exhibit slight interframe scene motion. The scene motion is removed and the segmentations of the two frames are superimposed in Figure 27. Note the alignment of the stationary targets in the foreground and the movement of the objects in the background. Magnified views of these targets are shown in Figures 28, 29, and 30. A stationary tank from the foreground is shown in Figure 28, while a moving APC and a tank are shown in Figures 29 and 30, respectively.

These examples have demonstrated that precise object position tracking can be achieved even when the platform and sensor are moving rapidly as in the AAH, RPV, and CV applications. These examples also illustrate the power of the approach in detecting minute relative target motion in the presence of extreme platform motion.



(a)



(b)

Figure 24. Magnified View of Aligned Clutter Object Outlines in Figure 23

Figure 25. Magnified View of Moving Tank Outline in Figure 23

(a)

(b)

Figure 26. Two Successive FLIR Frames From a Stationary Platform

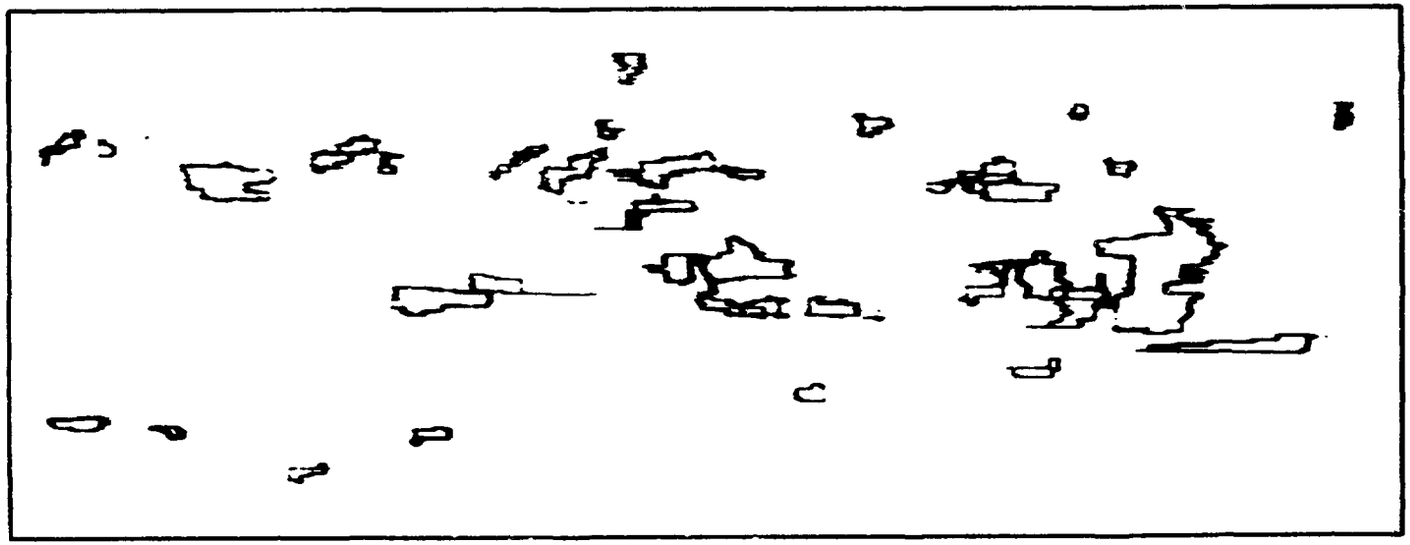


Figure 27. Object Outlines From Figure 26 Shown Superimposed, After Object-Matching and Compensating for Scene Motion. Note the alignment of the stationary targets.

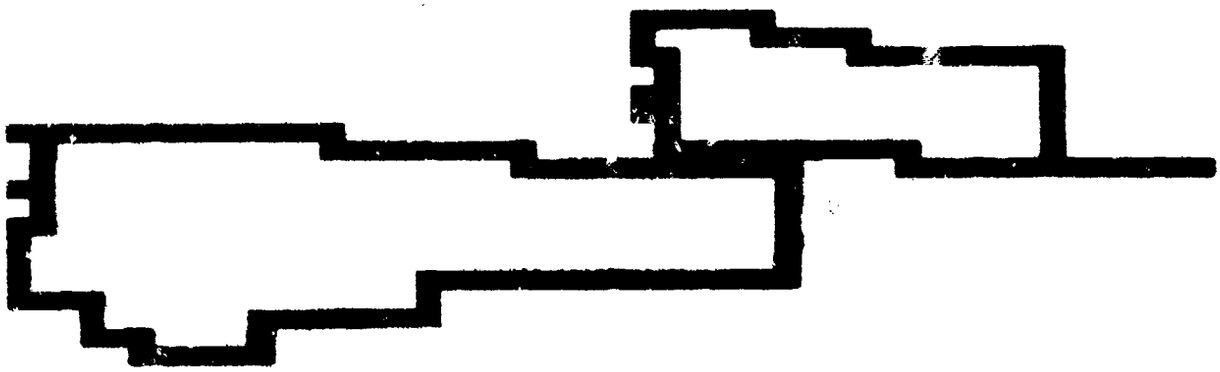


Figure 28. Magnified View of Stationary Tank in Figure 27, After Frame Alignment. Note precise registration.

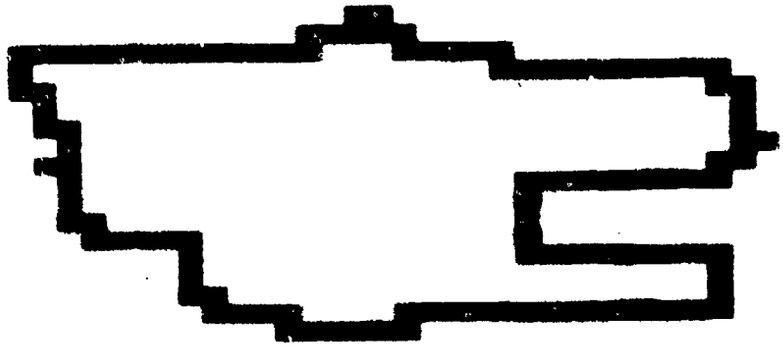


Figure 29. Magnified View of Moving APC in Figure 27

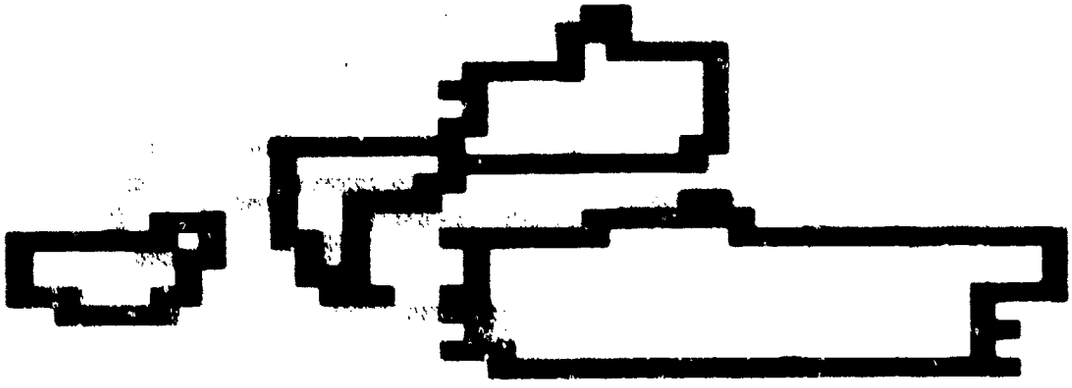


Figure 30. Magnified View of Moving Tank in Figure 27

SECTION VI
DATA BASE

This section summarizes the continuing tracking data base generation effort. An extensive FLIR video tape library of tactical targets in various backgrounds exists at Honeywell, acquired from NV&EOL and other sources under the current program and several others. Our approach to the selection and digitization of sequences for the simulation effort in this program will continue to be evolutionary. As each algorithm or subsystem is developed, we select image sequences which contain the features required for its evaluation. For example, the two FLIR sequences which have been digitized to date contain multiple moving targets with occlusion, from a stationary platform and a moving target from a fast moving platform. These have served to test the platform motion estimation and multiobject precision tracking capabilities. One of our next sequences will contain maneuvering targets, to test the object dynamics estimator to be developed next.

Following is a partial description of our video tape library containing the FLIR image sequences we have digitized to date.

The video tape data base for the verification of the algorithms we have discussed consists of six video tapes from FLIR and visual sensors. The six tapes contain interesting homing and tracking scenarios which will exercise all aspects of the tracking algorithm. The following paragraphs describe the different data sources and show examples of the imagery they contain.

NV&EOL "FORT POLK" SELECTIONS

This 525-line video tape was taken from a FLIR sensor mounted on a ground platform. Contained in this tape are examples of multiple moving and maneuvering targets in a high-clutter environment. Figure 31 shows a sequence of 11 frames which have been digitized. The entire sequence represents approximately 2 seconds. Note that some targets are occluded by the trees in the background and by the other targets in the image. This video tape also contains several examples of small long-range moving targets (from a wide field of view).

HIGH-PERFORMANCE AIRCRAFT SELECTIONS

This 525-line video tape was taken from a common-module FLIR sensor mounted on a high-performance aircraft. This tape provides examples of extreme sensor and platform motion. Figure 32 shows a sequence of eight digitized frames from this tape. Note the changes caused by platform motion and also the number of clutter objects in the scene. This sequence also contains a moving target.

NV&EOL FLIR TRACKING DATA BASE

These two 525-line video tapes of common module FLIR data contain both moving and stationary targets at various ranges. These tapes also provide some examples for the homing scenario. Two frames from this tape are shown in Figure 33.

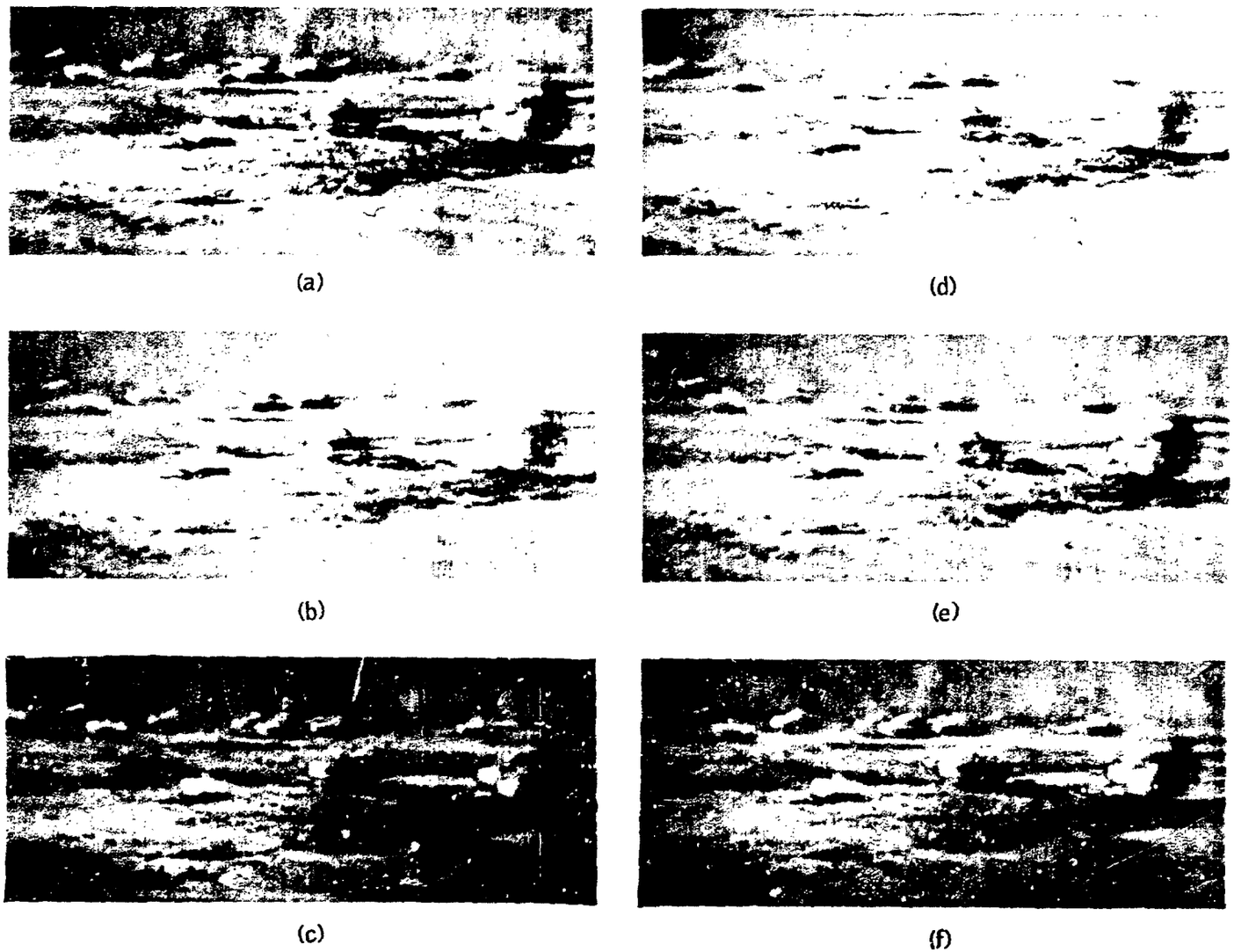


Figure 31. Example of "Fort Polk" Selections (Digitized)

(g)

(j)

(h)

(k)

(i)

Figure 31. Example of "Fort Polk" Selections (Digitized)--
Concluded

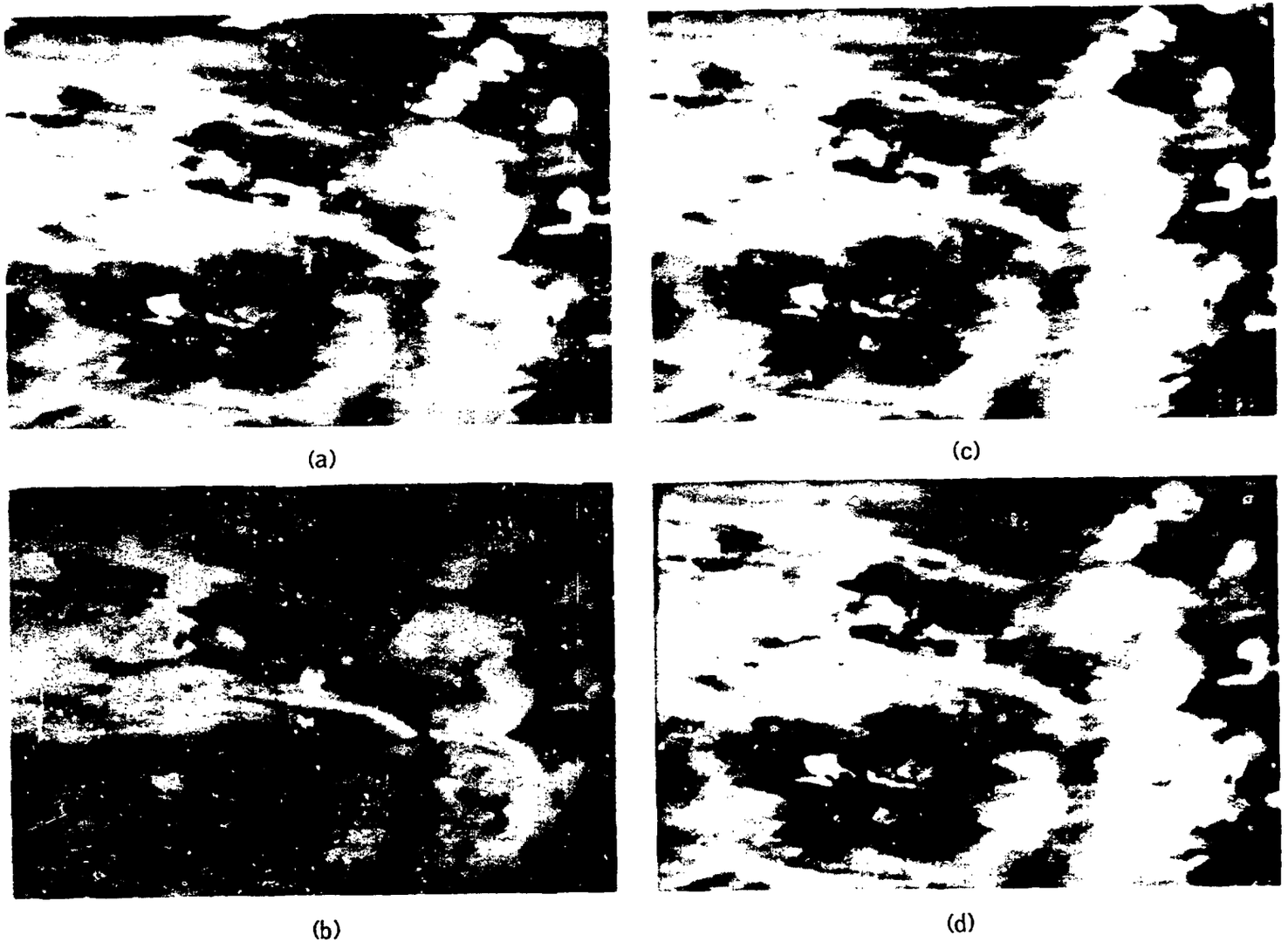


Figure 32. High-Performance Aircraft Selections

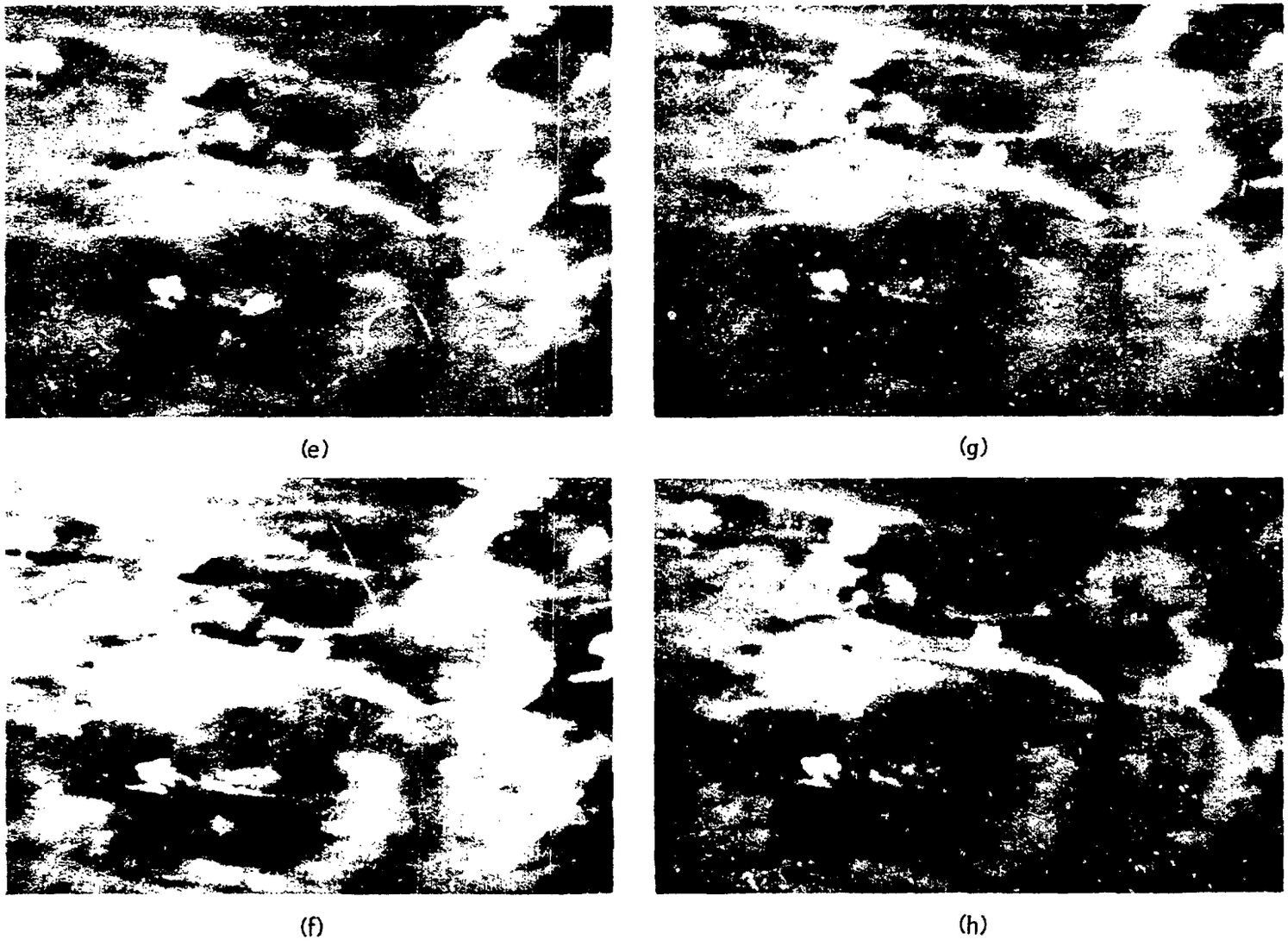


Figure 32. High-Performance Aircraft Selections -- Concluded

(a)

(b)

Figure 33. NV&EOL FLIR Tracking

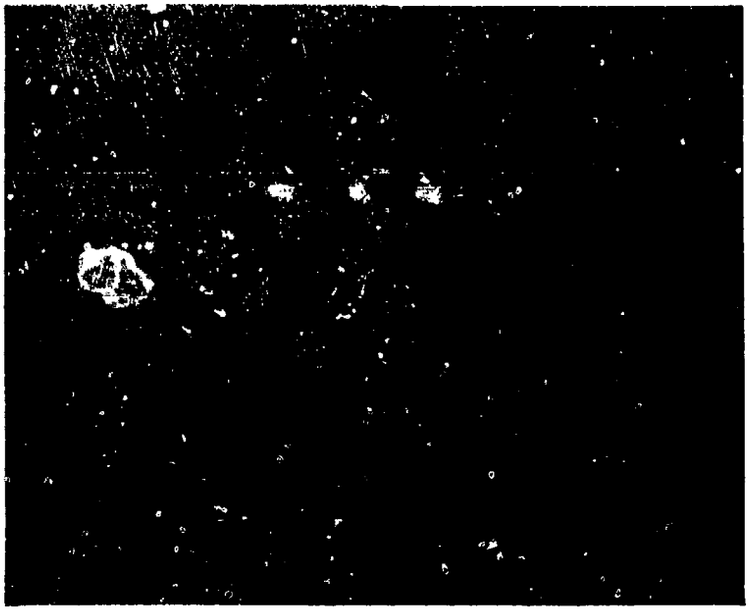
PATS TRAINING DATA BASE

This is an 875-line video tape from a common-module FLIR sensor mounted in an air platform. The targets from this tape include tanks and APCs from several aspect angles. Examples of this tape are shown in Figure 34.

NV&EOL TV TRACKING TAPE

This is a 525-line tape taken from a helicopter mounted TV sensor. This tape contains examples of gross sensor motion and multiple moving targets. In some sequences the targets leave and reenter the FOV because of the sensor platform instability. This tape also contains some good examples of the homing scenario.

(a)



(b)

Figure 34. PATS Training Data Base

SECTION VII
PLANS FOR THE NEXT REPORTING PERIOD

This section outlines the program plans for the next reporting period. Emphasis will be on:

- PATS simulation transfer
- Object-matching algorithms
- Scene model
- Background prediction techniques
- Advanced target recognition techniques
- Homing algorithms
- Data base generation

PATS SIMULATION SOFTWARE TRANSFER

The PATS simulation software, which has been converted to the EIKON image-handling formats, will be installed on the NV&EOL IBM 360 facility in this reporting period.

OBJECT MATCHING ALGORITHMS

Evaluation of the Fast Silhouette-Matching Algorithm will continue to characterize its performance and validate it with new data.

In particular, the number of iterations required and performance as a function of the initial position will be determined.

SCENE MODEL

The scene model which now contains only the platform displacement estimator will be extended in numerous ways. The scene motion model will be extended to include multiple frames from the current model which uses frame pairs. This implies matching objects, not only from the previous frame, but also from the preceding several frames. Inference techniques will be developed to associate multiple segments of the same object, for comparison with the new images. Using Kalman filter techniques, the apparent motion of both the objects and the platform derived from successive frames will be filtered over several frames to derive robust estimates of the platform and target motion dynamics. Object occlusion prediction will also be incorporated in the scene model.

BACKGROUND PREDICTION

Techniques for characterizing backgrounds (e.g., texture) will be developed. This will be incorporated in the motion-enhanced segmentation scheme to improve its performance, using the predicted target/background signatures.

ADVANCED TARGET RECOGNITION TECHNIQUES

Single frame target recognition algorithms developed under PATS will be improved to use the information from multiframe object matching in two ways -- first, through the accumulation of the decisions over several frames and computing the a posteriori probabilities; and second, through moving-target detection using target motion computed by the precise object-matching techniques.

HOMING TECHNIQUES

This includes the development of algorithms for critical aimpoint selection. Syntactic component recognition techniques developed under the "Automated Imagery Recognition System"* program will be applied to homing sequences to recognize target components and hence critical aimpoint selection.

DATA BASE

The preliminary data base will be expanded by digitizing new sequences from our video tape library which contains maneuvering targets, target occlusion, and varied background signatures. The development of the data base will run parallel with the development of the algorithms as the need arises for representative scenes for evaluating the algorithm.

---

*DARPA Contract No. F33615-76-C-1324.

REFERENCES

1. Reischer, B., "Assessment of Target Tracking Techniques," Proc., SPIE, pp. 67-71, Vol. 178, Smart Sensors, 1979.

2. Soland, D. and Narendra, P., "Prototype Automatic Target Screener," Ibid, pp. 175-184.

3. Soland, D., et al., "Prototype Automatic Target Screener, Goals and Implementation," U.S. Army Missile Command Workshop in Imaging Tracker and Autonomous Acquisition," November 1979.

4. Willet, T. and Raimondi, P.K., "Intelligent Tracking Techniques - A Progress Report," Proc., SPIE, pp. 72-75, Vol. 178, Smart Sensors, 1979.